PolyPublie
Polytechnique Montréal

POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

Titre: Title:	Système de reconnaissance de séquences musicales fredonnées utilisant un réseau à écho
Auteur: Author:	Corentin Faucher
Date:	2007
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Faucher, C. (2007). Système de reconnaissance de séquences musicales fredonnées utilisant un réseau à écho [Mémoire de maîtrise, École Polytechnique de Montréal]. PolyPublie. https://publications.polymtl.ca/7978/

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/7978/
Directeurs de recherche: Advisors:	
Programme: Program:	Non spécifié

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal

https://publications.polymtl.ca

UNIVERSITÉ DE MONTRÉAL

SYSTÈME DE RECONNAISSANCE DE SÉQUENCES MUSICALES FREDONNÉES UTILISANT UN RÉSEAU À ÉCHO

CORENTIN FAUCHER
DÉPARTEMENT DE GÉNIE ÉLECTRIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(APPRENTISSAGE MACHINE)
AVRIL 2007

© Corentin Faucher, 2007.

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 978-0-494-29231-0
Our file Notre référence
ISBN: 978-0-494-29231-0

NOTICE:
The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:
L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

SYSTÈME DE RECONNAISSANCE DE SÉQUENCES MUSICALES FREDONNÉES UTILISANT UN RÉSEAU À ÉCHO

présenté par: FAUCHER Corentin

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de:

M. CARDINAL Christian, Ph.D., président

M. BRAULT Jean-Jules, Ph.D., membre et directeur de recherche

M. GAGNON Michel, Ph.D., membre

À Pyranya

REMERCIEMENTS

Je souhaiterais remercier tout d'abord mon directeur de maîtrise, M. Jean-Jules Brault, de son appui et de ses précieux conseils dans la réalisation de ce travail. Je lui dois de plus la découverte de nombreux sujets de recherche passionnants. Je tiens à remercier également le président, M. Christian Cardinal, et le membre du jury, M. Michel Gagnon, pour leur participation à l'examen de ce mémoire.

Je ne voudrais pas oublier ceux et celles qui ont accepté de chanter lorsque je testais mon programme.

Je voudrais remercier enfin mes parents, ma fiancée et ses parents ainsi que mes amis de leur soutien. Leurs encouragements et leur aide m'ont été des plus précieux.

RÉSUMÉ

La recherche d'une pièce musicale dans une base de données se fait habituellement par le titre de la chanson ou le nom de l'auteur. Cependant, il peut arriver parfois qu'un air de musique nous trotte dans la tête sans que l'on puisse en donner le titre ou le nom de l'auteur. À ce moment-là, il serait opportun de pouvoir trouver la pièce musicale en fredonnant directement à l'ordinateur l'air que nous avons à l'esprit.

Cette recherche a pour but de déterminer une méthode efficace de stockage et de récupération de thèmes musicaux par le chant. Cette récupération se basera sur la similitude d'une mélodie chantée par un usager avec les différentes chansons contenues dans une collection de chansons.

Pour ce faire, on représentera les mélodies par des suites d'intervalles mélodiques (ratios de fréquences de deux notes successives) et de ratios d'intervalles temporels (ratios de durées de deux notes successives). Ceci permettra d'avoir une représentation invariante des mélodies à la tonalité et au tempo.

Dans ce mémoire, on a émis l'hypothèse que la proximité entre deux séquences mélodiques ne dépend pas uniquement du contour de la fréquence formé par les notes mais aussi de leur harmonicité. Pour ce faire, on a redéfini les intervalles mélodiques à l'aide d'une métrique dénommée « corrélation harmonique ». Celle-ci permet de décrire les intervalles mélodiques sous une forme qui respecte leur harmonie mutuelle. Par exemple, l'intervalle mélodique do-do est plus rapproché de l'intervalle do-sol que de l'intervalle do-ré. Ceci est dû aux harmoniques spectrales partagées entre les sons do, sol et ré.

Afin de décrire la distance entre différentes séquences musicales, on a choisi d'utiliser un réseau à écho. C'est un réseau de neurones (outils mathématiques) récurrent à mémoire évanescente possédant des propriétés de séparabilité et d'approximation. Pour chaque chanson inscrite dans la banque de chansons correspond un ensemble

d'états du réseau à écho. Ces états sont enregistrés dans une mémoire hétéro-associative ayant pour adresses les états du réseau à écho et pour données les numéros d'identification des différentes chansons.

Ainsi, pour retrouver une chanson chantée, on commence par identifier les notes, puis on stimule le réseau à écho avec les intervalles mélodiques et temporels entre les notes. On cherche ensuite dans la mémoire hétéro-associative la chanson qui se rapproche le plus des états générés par le réseau à écho.

Le système a été testé en employant une banque de 36 chansons. Une quinzaine de participants ont chanté librement une chanson de leur choix dans un premier temps, puis ont fait des « ta ta ta » et ont sifflé. Des 130 extraits fournis par les participants, on a obtenu un indice de classement MRR[1] (valeur moyenne de $1/rang$ des chansons demandées) de 76 % pour le mode chanté, 82 % pour les « ta ta ta » et de 85 % pour le mode sifflé. On a, de plus, démontré la capacité du système à reconnaître des variantes mélodiques d'une même chanson. Cependant, puisque les extraits recueillis ne comportaient pas de variantes mélodiques, le système était plus efficace lorsque l'on ne considérait pas la corrélation harmonique. Finalement, on a démontré la capacité du système à reconnaître un extrait débutant et se terminant à des moments arbitraires à l'intérieur d'une chanson.

Le premier chapitre de ce mémoire concerne les notions musicales de base qui seront utiles pour la compréhension des chapitres suivants. On y décrit les notes musicales, les intervalles et les regroupements musicaux.

Le deuxième chapitre décrit le prétraitement d'un extrait sonore. Celui-ci est particulièrement important, car il s'avère que la segmentation des notes est le principal problème lorsque l'on traite un signal brut provenant d'un locuteur humain.

Le troisième chapitre décrit la méthode utilisée pour représenter les intervalles mélodiques afin qu'ils respectent leur harmonie mutuelle.

Le quatrième chapitre traite de l'analyse de séquences au moyen du réseau à écho.

[1]Mean reciprocal rank (MRR).

Le programme de reconnaissance de séquences musicales réalisé utilise ce type de réseaux de neurones pour caractériser les séquences. On y décrit sommairement les réseaux de neurones et le cas particulier que sont les réseaux à écho.

Le cinquième chapitre traite de la recherche de séquences associées à l'aide de la mémoire associative de type hétéro-associative. Celle-ci a été utilisée avec le réseau à écho afin de trouver dans la collection de chansons celle ressemblant le plus à la chanson ayant été chantée par le locuteur. Dans ce chapitre, on traite également des techniques de recherche des plus proches voisins dans les espaces de haute dimension. Pour ce faire, on utilise les méthodes de recherche par arborescence (arbres métriques, spill-tree).

Finalement, le sixième chapitre décrit le programme de reconnaissance de séquences musicales et les résultats obtenus.

ABSTRACT

Song queries in databases usually rely on song title or author name for retrieval, but it can happen that someone would like to find a tune he has in mind of which he does know neither the title nor the author. In this case, it should be more appropriate to hum the tune directly to the computer so that the computer can find out which song it corresponds to.

This research aims to find an efficient way of querying musical songs by humming. The retrieval is made based on the similarity level between the query and the songs of the database.

We represent a melody with a sequence of melodic interval (pitch ratios of two successive notes) and inter onset intervals ratios (IOIr). This makes the melody both transposition invariant and tempo invariant.

In this thesis, we emit the hypothesis that the proximity between two melodies does not only rely on pitch contour but also on melodic aspect of notes sequences. To respect this melodic aspect we redefine the melodic intervals using a metric named "harmonic correlation". This correlation allows us to represent the melodic sequence in a way that respects the melodic content of the sequence. For example, the C-C interval is closer to C-G than to C-D. This is caused by the spectral harmonics shared between the notes C, D and G, respectively.

To describe the temporal structure of the databank melodies we choose an Echo State Network (ESN). This is an artificial neural network that has the properties of separation and approximation which make it useful for pattern classification. Each song in the database is associated with a set of internal states of the ESN. We saved these states in a hetero-associative memory. This hetero-associative memory got the internal states of the ESN for addresses and the song titles for associated data.

To retrieve a hummed song we start by identifying the musical notes. We then

stimulate the ESN with the melodic intervals and IOIr (Inter Onset Interval ratios). Finally we look for with the hetero-associative memory which song of the database is closest to the hummed query.

The query program has been tested using a database of 36 songs. Fifteen singers have hummed different songs in three different modes : free, "tatata" and whistle. 130 clips were acquired from the singers. We obtain a mean reciprocal rank or MRR (average value of 1/rank of the query) of 0.76 for the free mode, 0.82 for the "tatata" mode and 0.85 for whistle. We also demonstrate the ability of the program to recognize melodic variations of a song and its ability to recognize a query clip starting and ending at arbitrary times of the target song.

The first chapter of this thesis concerns musical notions which are used throughout the work. It describes the musical notes, musical intervals and sets of notes.

The second chapter describes the preprocessing of the acquired voice signal. This is very critical. If the musical notes are not well evaluated, the system performances will drop dramatically.

The third chapter describes the representation used to express the melodic intervals so they could respect the melodic aspect of the hummed clip.

The fourth chapter treats the musical sequence analysis by using an echo state network (ESN). This chapter briefly describes artificial neural networks and then describes the echo state networks and shows how it can be used for songs queries.

The fifth chapter treats the query method by using a hetero-associative memory. This memory has been used together with the ESN to find out in the databank which song is the most likely to be the hummed song. We also describe in this chapter the nearest neighbours searching techniques in high-dimension spaces using metric-tree and spill-tree.

Finally, the last chapter describes the program used for song retrieval and gives the experimental results.

TABLE DES MATIÈRES

LISTE DES FIGURES

LISTE DES NOTATIONS ET DES SYMBOLES

Notations

DTW	Association de contours, dynamic time warping
ESN	Réseau à écho, echo state network
HMM	Modèles de Markov cachés, hidden Markov models
IM	Intervalle mélodique
IOIr	Ratio de durées de notes, inter onset interval ratio
LPC	Prédiction linéaire, linear prediction coding
MIR	Recherche de l'information musicale, musical information retrieval
QBH	Recherche par fredonnement, query-by-humming
VIM	Vecteur d'intervalle mélodique

Symboles

$A(t)$	Amplitude de l'extrait sonore
$\mathbf{A}$	Ensemble des adresses de la mémoire hétéro-associative (états du réseau à écho)
A_{min}	Amplitude minimale du signal sonore
$\mathbf{C}$	Ensemble du contenu de la mémoire hétéro-associative (titres des chansons)
$comb()$	Fonction peigne de Dirac
$d()$	Distance entre deux nœuds
D	Nombre de subdivisions par octave, 12 dans l'échelle chromatique
$\mathbf{D}$	Ensemble des descendants d'un noeud (feuilles)
E	Erreur globale
$f(t)$	Fréquence fondamentale en fonction du temps au long de l'extrait sonore
f_{ech}	Fréquence d'échantillonnage, 22.05kHz

f_{dec}	Fréquence de découpage de l'extrait sonore, 43Hz
f_{ref}	Fréquence de référence, la 440Hz
$H(r)$	Fonction de corrélation harmonique
H	Matrice de corrélation harmonique
L	Nombre total de tranches de découpage
M	Espacement entre les tranches de découpage, 512 échantillons
M	Nombre d'intervalles mélodiques
N	Nombre de neurones internes du réseau à écho
N	Nombre d'échantillons par tranche de découpage, 1024 échantillons
n	Temps discret
$\mathbf{n}$	Nœud d'un arbre de recherche
r	Intervalle mélodique en demi-tons
R_{min}	Distance minimale entre deux nœuds
S	Nombre total d'échantillons
t	Temps continu
$T_{1/2}$	Demi-vie du réseau à écho
$T()$	Opérateur de l'impact d'une séquence d'entrée sur le réseau à écho
$\vec{u}_r$	Vecteur d'intervalle mélodique
$\vec{u}(n)$	Entrée du réseau à écho au temps n
$\bar{\mathbf{u}}$	Historique des entrées du réseau à écho
$\vec{x}_l$	Tranche de l'extrait sonore avec fenêtre de Hamming
$\vec{x}(n)$	État interne du réseau à écho au temps n
$\bar{\mathbf{x}}$	Historique des états internes du réseau à écho
X	Matrice N par L contenant toutes les tranches de l'extrait sonore.
U	Matrice des vecteurs d'intervalle mélodique de taille N x M
$\mathbf{W}$	Matrice d'interconnexion des neurones internes du réseau à écho
$\mathbf{W}^{in}$	Matrice des poids d'entrée du réseau à écho
$\mathbf{W}^{out}$	Matrice des poids de sortie du réseau à écho

∇	Gradient
γ	Variation maximale de la fréquence
Δf_{plt}	Hauteur des plateaux de fréquence
Δt_{min}	Durée minimale d'une note (anti-rebond)
ε	Indice de classement des nœuds d'un arbre
$\nu(t)$	Signal binaire donnant la présence ou l'absence d'une note au long de l'extrait sonore
$\rho()$	Corrélation entre deux vecteurs
$\rho()$	Rayon Spectral d'une matrice
τ	Seuil de décision, 0,5
$\omega(n)$	Fenêtre de Hamming

LISTE DES TABLEAUX

LISTE DES ANNEXES

INTRODUCTION

Depuis quelques années, l'avancement des technologies en informatique a permis une véritable explosion de la disponibilité de la musique en ligne dans Internet. Ceci a amené une recherche importante sur les méthodes de récupération de pièces musicales. La plupart des moteurs de recherche actuels reposent sur l'utilisation de titres de fichiers ou d'étiquettes texte associées aux fichiers de musique pour leur recherche. On utilise peu de méthodes axées sur la reconnaissance de thèmes musicaux. Et pourtant, il arrive que l'on n'ait aucune idée du titre ou de l'auteur d'une chanson qui nous trotte dans la tête. Il serait alors opportun de rechercher la pièce musicale en fredonnant ou en sifflant un extrait de cette pièce.

Nous avons donc à faire face à un problème de reconnaissance de séries temporelles formées par des séquences de notes. Plusieurs possibilités s'offrent à nous pour identifier ce type de séquences. La première approche est de nature statistique. Dans celle-ci, on considère la nature probabiliste de l'information que l'on cherche à reconnaître. Les modèles probabilistes souvent utilisés dans ce type de problème sont les modèles de Markov[2]. Dans ces modèles, on suppose que la séquence est modélisée par un processus aléatoire respectant certaines règles. On peut alors déterminer les paramètres du modèle qui expriment le mieux la séquence observée.

Une autre approche consistant à reconnaître les motifs, utilisée dans le cadre de la reconnaissance de mélodies, est l'association de contours. Dans ce cas, on cherche à aligner le tracé formé par les notes ayant été chantées avec les tracés formés par les notes des mélodies contenues dans la banque de données (voir figure 1). Il s'agit en fait d'une méthode de programmation dynamique dénommée « elastic matching » ou « dynamic time warping » (DTW). Cette méthode permet de

[2]Hidden Markov models (HMM).

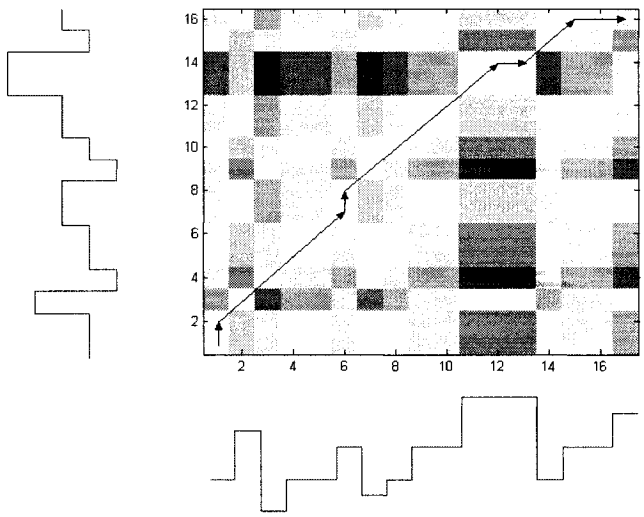

FIG. 1 Chemin de moindre coût pour aligner deux séquences.

déterminer laquelle parmi les séquences de la banque de données présente le coût minimum d'ajustement afin de trouver la séquence de référence.

Plusieurs recherches ont déjà été effectuées dans le domaine de la reconnaissance de séquences musicales. Ce domaine de recherche est parfois identifié par l'acronyme QBH, pour « query-by-humming ». On trouve également le terme MIR pour « music information retrieval ». Le projet MUSART (Dannenberg et al., 2003), réalisé en collaboration avec les universités du Michigan et de Carnegie Mellon, est un bon exemple d'efforts faits en ce sens. Les chercheurs ont développé des algorithmes basés sur les modèles de Markov et l'association de contours de mélodies pour reconnaître des séquences de notes. Ils ont également réalisé une banque de mélodies de 2 844 thèmes extraits de 258 chansons des Beatles prises directement dans les fichiers MIDI. Leur système a permis de mettre en évidence la diminution de la qualité du classement de la chanson demandée en fonction du nombre de mélodies dans la banque de données. Les chercheurs ont pu démontrer que l'indice de classement[3] d'une chanson donnée par un locuteur suit une loi $1/log(N)$ pour

[3]L'indice de classement utilisé dans cette étude et dans d'autres études de « query-by-

une banque de données contenant N chansons (plus il y a de chansons dans la banque de données, moins bonnes sont les performances). D'autres travaux ont été effectués tels que ceux de Norman H. Adams (Adams, 2004) pour l'extraction des notes à partir du signal sonore par programmation dynamique et à l'aide de HMM entre autres. Dans le cas de Norman H. Adams, il a utilisé 14 mélodies initiales et s'est créé une banque de 3 570 thèmes synthétiques. On peut, de plus, noter le système de reconnaissance de mélodies de l'Institut Fraunhofer[4] ainsi que celui de l'Université de New York[5].

De manière générale, soit que les systèmes comparent directement le contour de la fréquence dans le temps $f(t)$ fournie par un usager avec les contours de la banque de données ou qu'ils extraient d'abord les notes ayant été chantées, puis cherchent ensuite à comparer la séquence des notes recueillies avec les séquences enregistrées dans la banque de données (soit avec les HMM, soit avec l'association de contours). Dans cette recherche, on extraira d'abord les notes de la séquence enregistrée par l'usager, puis l'on tentera de déterminer la séquence qui lui ressemble le plus dans la banque de données.

Dans le cadre de ce mémoire, on a décidé d'essayer de nouvelles techniques pour reconnaître les séquences musicales. On a d'abord émis l'hypothèse que la proximité entre deux séquences mélodiques ne dépend pas uniquement du contour de la fréquence fondamentale des sons en fonction du temps mais aussi de leur harmonicité. L'harmonie est en quelque sorte le cadre d'une chanson. Si les notes d'une chanson sont modifiées, mais qu'elles demeurent dans le cadre harmonique, la mélodie reste semblable. Pour tenir compte de l'harmonie, on a redéfini les intervalles mélodiques (distances entre les notes) à l'aide d'une métrique dénommée « corrélation

humming » est l'inverse du taux de classement moyen et est dénoté MRR pour « mean reciprocal rank » (voir chapitre 6).

[4] *http : //www.idmt.fraunhofer.de/eng/research_topics/query_by_humming.htm*

[5] *http : //querybyhum.cs.nyu.edu/*

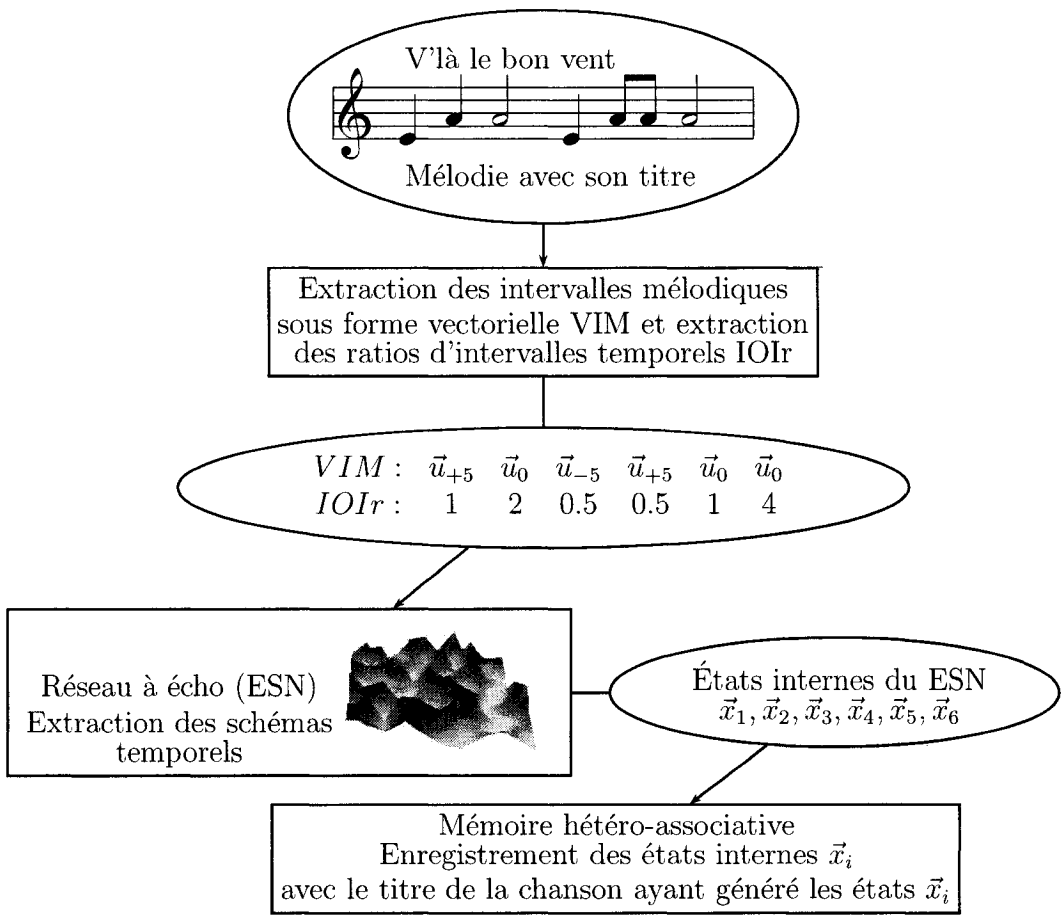

FIG. 2 Schéma bloc de l'entraînement du système de reconnaissance de séquences musicales.

harmonique ». Celle-ci permet de représenter dans un nouvel espace les intervalles mélodiques sous une forme qui respecte leur harmonie mutuelle.

Ensuite, au lieu d'utiliser une des approches classiques (la programmation dynamique ou les modèles de Markov cachés) pour reconnaître les séquences, nous avons utilisé un réseau de neurones récurrent particulier développé par H. Jeager (Jaeger, 2001) qu'est le réseau à écho ou ESN[6]. Il s'agit d'une architecture possédant une mémoire évanescente permettant l'émergence des propriétés de séparabilité et d'ap-

[6]Echo state network (ESN).

proximation, ce qui en fait un excellent outil pour la reconnaissance de séquences musicales.

Finalement, dans notre étude, nous avons utilisé une nouvelle méthode d'analyse des divers états du réseau à écho. Au lieu d'entraîner des neurones de sortie à reconnaître des motifs d'états, nous avons utilisé une mémoire associative associant directement les états du réseau à écho aux titres des chansons.

Le premier chapitre du mémoire introduit les notions musicales de base. Ces dernières seront très utiles pour décrire les notions données tout au long du travail.

On abordera ensuite le problème du prétraitement du signal. On peut observer, à la figure 3, le schéma bloc pour la reconnaissance d'une séquence. Après l'acquisition du signal brut, ayant été chanté par un locuteur, on procède au prétraitement du signal. Celui-ci permet d'extraire les notes ayant été chantées par un locuteur. On y traite donc du problème de segmentation des notes chantées. La série de notes obtenue sera ensuite représentée par une suite d'intervalles mélodiques (distance entre deux notes), notés IM, et de ratios de durées de notes ou IOIr pour "inter onset interval ratio". Cette représentation permet de garder la mélodie invariante par rapport à la tonalité et au tempo.

Avant d'aborder la reconnaissance de séquences musicales, nous chercherons une meilleure métrique afin de représenter les intervalles mélodiques. Ainsi, comme nous le verrons dans le chapitre 3, l'important est de respecter la mélodie générale et non pas de chanter les notes de la mélodie. C'est pour cette raison que l'on cherchera à définir la corrélation harmonique. Celle-ci permet de redéfinir les intervalles mélodiques en vecteurs d'intervalle mélodique ou VIM (voir figures 2 et 3). Cette nouvelle définition des intervalles mélodiques permettra de mieux décrire l'aspect

mélodique des séquences musicales qui seront alors décrites par une série de couples $\langle VIM, IOIr \rangle$.

Puis, à l'aide d'un réseau à écho ou ESN (réseau de neurones récurrent particulier), la suite mélodique sera analysée comme un signal. On fournira à chaque instant les différents couples $\langle VIM, IOIr \rangle$ de la séquence mélodique pour stimuler ce réseau (voir figures 2 et 3). Les ESN possèdent une mémoire évanescente et mémorisent ainsi de façon implicite des structures temporelles de courte durée dans leur état interne. Le réseau à écho est donc tout à fait approprié pour modéliser des séquences musicales en caractérisant les structures musicales de courte durée que sont les thèmes musicaux.

Dans le chapitre 5, on décrit la mémoire associative de type hétéroassociative qui sera utilisée avec le ESN afin d'associer aux états internes générés par le réseau à écho le titre d'une chanson. Dans la phase d'entraînement, on écrit dans cette mémoire associative les états internes générés avec le titre de la chanson de la collection (voir figure 2). Dans la phase de reconnaissance, on cumule les chansons retenues par la mémoire associative pour chaque état interne généré par le ESN et on affiche les résultats dans l'ordre croissant (voir figure 3).

Finalement, le dernier chapitre décrit le programme de reconnaissance de séquences musicales et donne les résultats qu'il a permis d'obtenir. Une première version du programme réalisé permet d'effectuer les différents tests sur les extraits de chansons recueillis afin d'optimiser les paramètres du système. Une seconde version réalisée comporte une interface usager. Cette version permet à un chanteur d'effectuer l'acquisition d'un extrait musical. Il peut par la suite visualiser et écouter le résultat, corriger les notes recueillies, effectuer une recherche de son extrait. Il lui est également possible de faire jouer les mélodies de la collection de chansons.

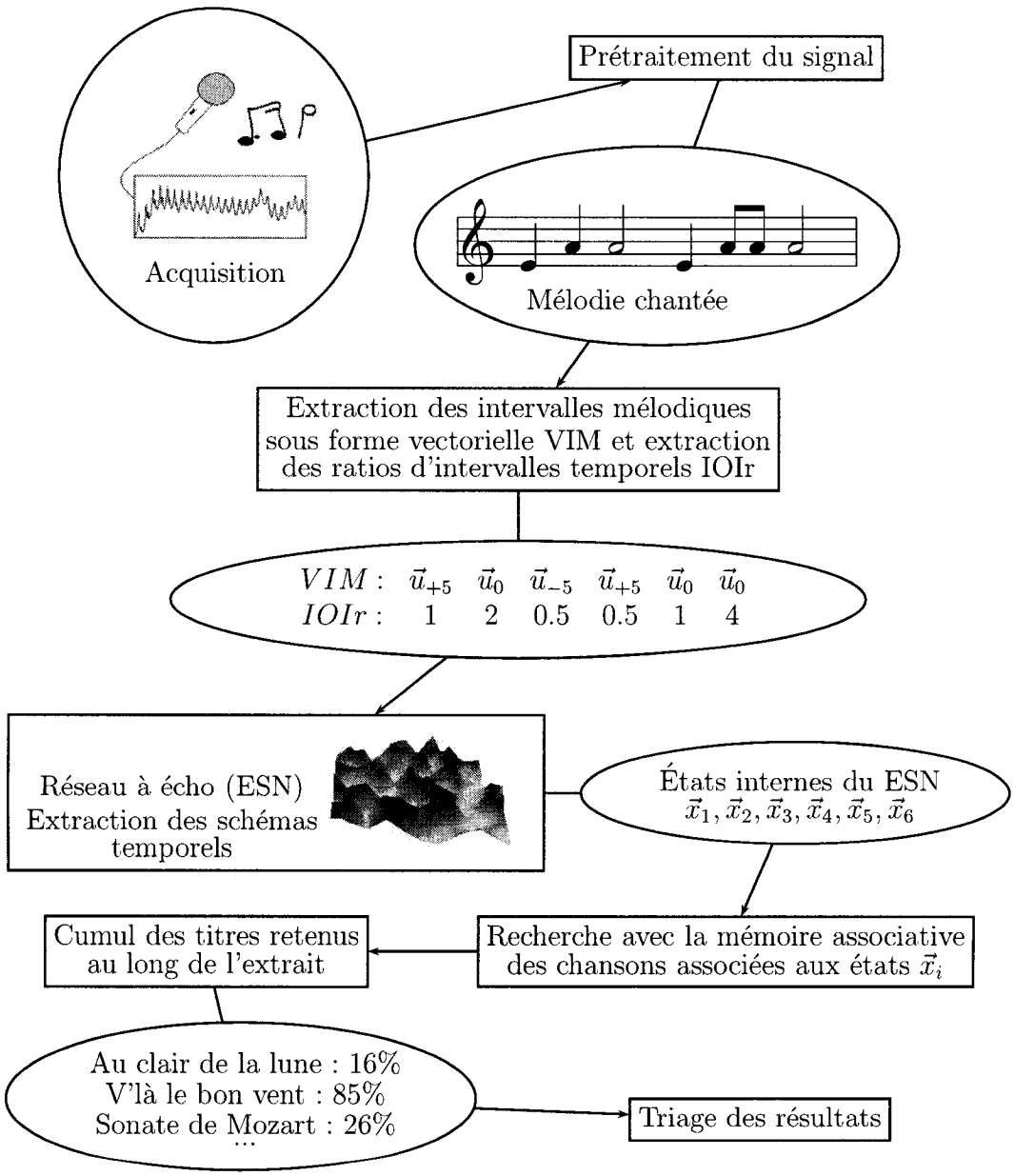

FIG. 3 Schéma bloc de la phase de reconnaissance d'une séquence musicale.

CHAPITRE 1

NOTIONS MUSICALES

Avant d'aborder l'analyse de séquences sonores, il est essentiel d'avoir une bonne connaissance de la théorie musicale. Dans ce chapitre, nous aborderons différents concepts musicaux de base tels que les notes musicales, les intervalles, les groupements musicaux et le timbre d'une note.

1.1 Notes musicales

En premier lieu, il y a les notes. Elles permettent de composer la musique. Chaque note est caractérisée par une fréquence qui lui est propre : sa fréquence fondamentale. Les fréquences fondamentales associées aux notes musicales ne sont pas distribuées selon une échelle linéaire. Elles suivent plutôt une échelle logarithmique. En effet, dans la gamme diatonique, c'est l'octave qui caractérise le cycle des notes. L'octave étant la distance entre deux notes pour laquelle la fréquence double. Il s'agit de plus de l'intervalle dans lequel le cycle des notes se répète (un octave au-dessus de la note do central, on retrouve le do aigu). Ainsi, dans la gamme diatonique en do majeur, on retrouve les huit notes : do, ré, mi, fa, sol, la, si, do. Ces notes sont, pour la plupart, distantes d'un ton, à l'exception des notes $[mi, fa]$ et $[si, do]$ qui sont distantes d'un demi-ton[1]. On compte donc 12 demi-tons par octave, ce qui correspond à l'échelle chromatique. En utilisant $D = 12 \in \mathbb{N}$ qui est le

[1]Dans la gamme diatonique majeure, on retrouve deux tétracordes : succession de quatre notes conjointes disposées dans l'ordre (un ton, un ton et un demi-ton). Ces tétracordes sont distantes d'un ton.

nombre de subdivisions dans un octave, on peut définir la fréquence fondamentale d'une note comme étant :

$$f_{note} = f_{ref}\ 2^{\frac{r_{note}}{D}} \tag{1.1}$$

La fréquence de référence f_{ref} est habituellement celle de la note $la_3 = 440Hz$, bien connue en musique. La variable $r_{note} \in \mathbb{N}$ donne le nombre de subdivisions entre la fréquence de référence et la note considérée.

Il est à noter qu'il existe d'autres échelles telles que l'échelle holdérienne[2] pour laquelle $D = 53$. Cependant, en ce qui concerne notre étude, comme la marge d'erreur dans l'évaluation des notes est de l'ordre du demi-ton, une telle précision ne sera pas nécessaire. En effet, il serait étonnant que les séquences chantées par un utilisateur le soient avec autant de précision. On se limitera donc à l'échelle chromatique. De plus, on ne considérera que les notes incluses dans l'intervalle $[la_0\ la_5]$, soit de $la_0 = 55Hz$ à $la_5 = 1.76kHz$ (voir tableau 1.1). Cela couvre amplement l'ensemble des notes émises par les locuteurs humains, y compris les notes sifflées plus aiguës.

1.1.1 Figures de notes

Une figure de note, c'est la forme d'une note (voir figure 1.1). Les figures de notes permettent de distinguer les différentes durées des notes. Tout comme la fréquence, la durée des notes varie de façon exponentielle. On observe d'abord la ronde dont la durée est de quatre temps[3]. Elle est suivie de la blanche dont la durée est de

[2] L'échelle holdérienne divise l'octave en 53 commas. Dans cette échelle, on compte neuf commas par ton et cinq par demi-tons. En considérant par exemple le $sol_2^\sharp$, on aurait, dans l'échelle diatonique : $f_{sol_2^\sharp} = 440Hz\ 2^{\frac{1}{12}} = 415.3Hz$, alors que dans l'échelle holdérienne, on aurait : $f_{sol_2^\sharp} = 440Hz\ 2^{\frac{4}{53}} = 417.6Hz$. Le résultat donne une petite différence d'environ $\frac{1}{10}$ de demi-ton pouvant être perçue par certains musiciens.

[3] Ici, les temps sont comptés en temps de noires. Cependant les temps sont relatifs et peuvent être comptés en employant la figure de notes désirée.

Note	Fréquence f_{note}	Intervalle r_{note}	
la_0	55.0 Hz	-36	
$la_0^\sharp$	58.3 Hz	-35	
...	...	...	
$sol_2^\sharp$	415 Hz	-1	
la_3	440 Hz	0	Unisson
$la_3^\sharp$	466 Hz	1	Seconde mineure
si_3	494 Hz	2	Seconde majeure
do_3	523 Hz	3	Tierce mineure
$do_3^\sharp$	554 Hz	4	Tierce majeure
re_3	587 Hz	5	Quarte juste
$re_3^\sharp$	622 Hz	6	4^{te} aug. / 5^{te} dim.
mi_3	659 Hz	7	Quinte juste
fa_3	698 Hz	8	Sixte mineure
$fa_3^\sharp$	740 Hz	9	Sixte majeure
sol_3	784 Hz	10	Septième mineure
$sol_3^\sharp$	831 Hz	11	Septième majeure
la_4	880 Hz	12	Octave majeure
$la_4^\sharp$	932 Hz	13	Octave augmentée
...	...	...	
$sol_4^\sharp$	1.66 kHz	23	
la_5	1.76 kHz	24	

TAB. 1.1 Tableau des notes et de leur fréquence, $D = 12$ et $f_{ref} = la_3 = 440Hz$.

deux temps. Vient ensuite la noire dont la durée est de un temps. Finalement, il y a les croches et les doubles croches qui ont respectivement une durée de $\frac{1}{2}$ et de $\frac{1}{4}$ de temps.

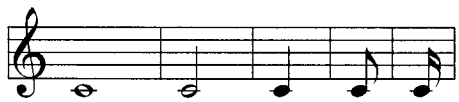

FIG. 1.1 Les différentes figures (ou périodes) de notes.

1.1.2 Timbre d'une note

Une autre notion musicale importante dans le cadre de ce travail est le timbre musical d'un son. Le timbre d'un son peut se définir de manière générale par l'ensemble des paramètres permettant d'identifier le type de son entendu. L'évolution temporelle du son peut être une première caractéristique du son. Ainsi, un son peut être plus sec ou plus entretenu. La présence de bruits transitoires peut également être un des attributs du timbre. Cependant, le point le plus important et sur lequel on portera le plus d'attention, c'est l'enveloppe spectrale.

1.1.2.1 Partiels d'un signal sonore

Pour mieux comprendre ce qu'est le timbre d'un son, il faut d'abord savoir le décomposer. Pour analyser un signal sonore $s(t)$, on le décompose généralement en une somme de signaux plus simples appelés partiels, qui sont notés $s_i(t)$. On représente donc un signal sonore $s(t)$ en une somme de partiels $s_i(t)$ auquel on ajoute un signal de bruit $b(t)$:

$$s(t) = \sum_i s_i(t) + b(t) \tag{1.2}$$

Le partiel d'un signal est par définition une composante sinusoïdale de ce signal. Il se caractérise par une fréquence f_i, une amplitude A_i et une phase initiale ϕ_i. Le partiel s'exprime donc sous la forme :

$$s_i(t) = A_i sin(2\pi f_i t + \phi_i) \tag{1.3}$$

Un signal composé de partiels possède un spectre comportant de nombreuses raies. Chaque raie est associée à un partiel de fréquence f_i. On donne, à la figure 1.2, un exemple de spectre d'un son de gong. Comme on peut l'observer, il contient de nombreux partiels identifiables par des pics.

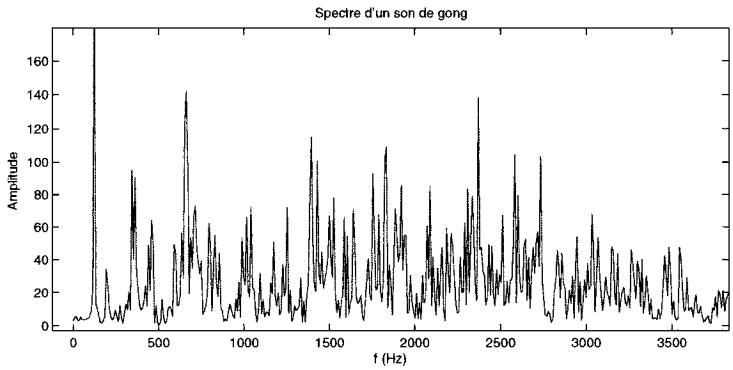

FIG. 1.2 Spectre d'un son de gong.

1.1.2.2 Enveloppe spectrale

Dans la plupart des instruments de musique, le son est d'abord produit par une source appelée excitateur (par exemple, les cordes vocales), puis il est mis en forme à l'intérieur d'un résonateur (par exemple, le conduit vocal). Le signal généré par l'excitateur se compose généralement d'harmoniques. Les harmoniques sont les partiels du son. Elles possèdent la caractéristique d'être des multiples entiers de la fréquence fondamentale. Par exemple, la note $la_3 = 440Hz$ possède comme première

harmonique le partiel ayant $f_1 = 440Hz$, puis comme deuxième, le partiel ayant $f_2 = 880Hz$ et ainsi de suite. L'amplitude des harmoniques décroît généralement avec le numéro d'harmonique.

Pour former un son, on utilise aussi un résonateur. Ce dernier agit comme un filtre favorisant certaines fréquences. Dans le spectre des fréquences, il s'agit d'une courbe continue, nommée enveloppe spectrale, épousant les pics des différents partiels. De plus, cette courbe varie dans le temps et traduit à chaque instant les rapports d'amplitude entre les différents partiels du son. Il s'agit donc d'un paramètre très important pour décrire le son.

On observe généralement que les instruments de musique (ou la voix) ne possèdent pas de spectre monochromatique, mais bien une grande quantité d'harmoniques spectrales. C'est d'ailleurs cette distribution des harmoniques (ou enveloppe spectrale) qui permet de distinguer les différents instruments de musique ou les différentes voyelles de la langue parlée.

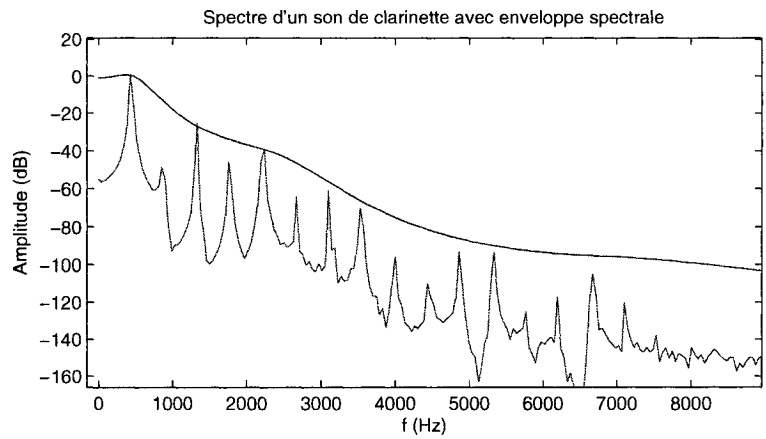

FIG. 1.3 Spectre et enveloppe spectrale d'un son de clarinette.

Une méthode couramment utilisée dans l'évaluation de l'enveloppe spectrale d'un son est la prédiction linéaire ou LPC[4] (Pachet, 2004; Rabiner et Juang, 1993). Dans

[4]Linear predictive coding (LPC).

ce modèle, on cherche le filtre qui représente le mieux l'aspect général du spectre du son. Ce dernier est estimé par un filtre autorégressif d'ordre p noté : $H(z) = G/\sum_{i=0}^{p} a(i)z^{-i}$. À la figure 1.3, on présente un exemple d'enveloppe spectrale avec $p = 10$. Plus l'ordre du filtre est grand, plus l'enveloppe est précise et épouse bien les pics d'harmoniques. Cependant, si l'ordre p est trop élevé, la prédiction linéaire créera une enveloppe trop précise produisant des creux entre les pics d'harmoniques.

1.2 Intervalles

À présent que l'on a défini le concept des notes musicales, on va définir l'intervalle entre deux notes. Un intervalle est défini comme la distance séparant deux notes et est noté r. Cette distance se compte en demi-tons dans l'échelle diatonique.

On distingue deux principaux types d'intervalles : l'intervalle harmonique et l'intervalle mélodique. L'intervalle harmonique est la distance entre deux notes émises simultanément, alors que l'intervalle mélodique est la distance entre deux notes successives. Dans la présente étude de reconnaissance de séquences musicales, nous nous intéresserons principalement aux intervalles mélodiques, c'est-à-dire que nous analyserons des suites de notes chantées par des locuteurs. Bien entendu, il n'est pas possible de chanter deux notes à la fois.

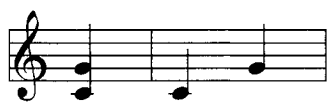

FIG. 1.4 Intervalle harmonique et intervalle mélodique.

Les différents intervalles sont illustrés à la figure 1.5. On trouve dans l'ordre : l'unisson[5], la seconde, la tierce, la quarte, la quinte, la sixte, la septième, l'octave,

[5]Bien que l'unisson ne soit pas toujours reconnu comme un intervalle (Vincent-d'Indy, 1982), nous le considérerons par commodité, dans la présente étude, comme un intervalle.

la neuvième, etc. On qualifie un intervalle comme ascendant lorsque la seconde note est supérieure à la première note et comme descendant dans le cas contraire.

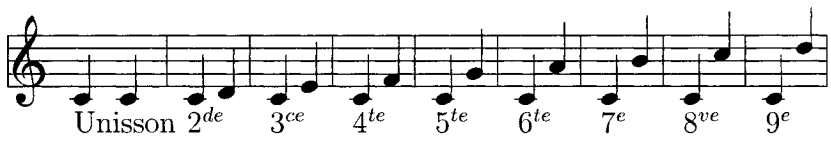

FIG. 1.5 Intervalles mélodiques.

Pour analyser une séquence musicale, on doit se fier à la séquence des intervalles mélodiques observés. En effet, c'est la suite des intervalles qui permet d'identifier un air de musique et non seulement la suite des notes particulières. Par exemple, il est possible de prendre une pièce en do majeur et de la transposer en mi majeur. Dans ce cas, la tonalité[6] change, mais l'air demeure le même.

L'intervalle mélodique r entre deux notes successives de fréquences fondamentales $f(t_1)$ et $f(t_2)$, émises respectivement aux temps t_1 et t_2, se définit à l'aide de la relation 1.1 comme étant :

$$\begin{aligned} r\left(f(t_1), f(t_2)\right) &= r\left(f(t_1)\right) - r\left(f(t_2)\right) \\ &= D log_2\left(\frac{f(t_1)}{f_{ref}}\right) - D log_2\left(\frac{f(t_2)}{f_{ref}}\right) \\ &= D log_2\left(\frac{f(t_1)}{f(t_2)}\right) \end{aligned} \tag{1.4}$$

Dans cette étude, les intervalles seront comptés en demi-tons ($D = 12$). Cette distance entre les notes peut être observée de façon continue à chaque instant t et permet de détecter les intervalles mélodiques. La différence de temps (Δt) entre deux observations dépend directement du tempo de l'air musical. On y reviendra plus tard, au chapitre 4 traitant de l'analyse des séquences.

[6]La tonalité ou le ton est l'ensemble des sons appartenant à une gamme diatonique. Par exemple, une mélodie dans la tonalité en sol majeur peut contenir les notes : sol, la, si, do, ré, mi et fa dièse.

1.3 Regroupements musicaux

Les concepts de base des mélodies musicales ayant été brièvement rappelés, on peut alors s'interroger sur la nature des structures musicales.

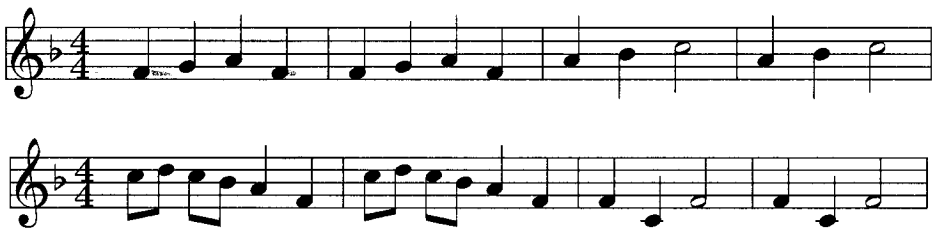

FIG. 1.6 Mélodie musicale de Frère Jacques

Comme on peut l'observer à la figure 1.6, une pièce musicale peut se décomposer en groupes de notes. Donc, après les intervalles viennent les groupements musicaux. Ainsi, lors de l'analyse d'une séquence musicale, la description que l'on peut faire de la séquence pourra se baser sur les groupes de notes. Ce sont eux qui forment la structure réelle d'une pièce musicale.

Lors de l'analyse de séquences musicales, on cherche ces regroupements de notes et on les identifie comme étant des schémas musicaux. Ils sont généralement de faible étendue temporelle. Dans l'extrait de Frère Jacques (figure 1.6), on observe quatre différents schémas se répétant : la 1^{re} mesure[7] se répète à la 2^{e}, la 3^{e} à la 4^{e}, la 5^{e} à la 6^{e} et la 7^{e} à la 8^{e}. Cependant, l'identification de ces schémas musicaux n'est pas toujours évidente. La détermination et la caractérisation des schémas musicaux composent d'ailleurs un des principaux objectifs de la musicologie.

[7]Une mesure est la division d'un morceau en sections d'égale durée séparées par la barre de mesure.

Pour l'identification de regroupements de notes ou de schémas musicaux, on se fie généralement aux trois principaux critères suivants (Lartillot, 2003) :

1. les groupements basés sur le style : Dans ce cas, on cherche à identifier les schémas en observant l'harmonie et la structure temporelle.
2. les frontières locales : La suite musicale peut être segmentée par trois types de discontinuités locales : un changement brusque dans l'intensité des notes, une grande variation dans la fréquence des notes, un intervalle temporel plus long entre les notes.
3. la répétition : Une expression musicale est souvent gérée par une reproduction de quelques schémas se répétant. En identifiant ces répétitions, on peut identifier les schémas de base d'une composition.

Dans ce projet, on cherche à reproduire ce que nous faisons tous inconsciemment lors de l'écoute d'une chanson, c'est-à-dire de la reconnaissance de schémas musicaux. Ce sont ces schémas de base, constituant des expressions musicales, qu'il faut reconnaître et classer. Aux chapitres 4 et 5, nous aborderons, de façon plus concrète, la caractérisation et l'identification des schémas permettant la reconnaissance d'une pièce musicale.

1.4 Conclusion du chapitre

Dans cette section, nous avons vu les principales notions musicales qui nous seront utiles pour décrire les séquences mélodiques. Cependant, avant d'analyser une séquence musicale proprement dite, il nous faut d'abord extraire les notes du signal sonore émis par un locuteur.

CHAPITRE 2

PRÉTRAITEMENT D'UN EXTRAIT SONORE

Afin de pouvoir identifier une pièce musicale lors de son écoute, il est important d'abord de détacher les notes d'un extrait sonore émis par un locuteur. Que ces notes soient sifflées, fredonnées ou chantées, l'opération consiste à trouver leur fréquence fondamentale et leur durée.

Cette détermination des notes est particulièrement critique, car les extraits sont chantés par des locuteurs humains. Ces derniers émettent un signal sonore qui n'est pas très régulier et qui peut s'avérer fort difficile à analyser. Les séparations entre les notes ne sont pas toujours claires. La fréquence d'une note émise par un locuteur peut varier durant le temps qu'elle est émise et, de plus, le tempo peut varier durant l'extrait.

Pour analyser un extrait sonore, on effectuera d'abord l'acquisition du signal sonore à un taux d'échantillonnage f_{ech}[1], à l'aide d'un microphone. Ensuite, ce signal devra être découpé en tranches temporelles de courte durée sur lesquelles on effectuera l'analyse nécessaire qui permettra d'extraire la fréquence fondamentale ainsi que l'amplitude fournie à chaque instant. Finalement, on identifiera les notes chantées en observant la variation de la fréquence fondamentale dans le temps. La figure 2.1 décrit de façon générale les différentes étapes du prétraitement du signal.

[1] Des valeurs typiques de taux d'échantillonnage sont de $11.025kHz$, $22.05kHz$ et $44.1kHz$.

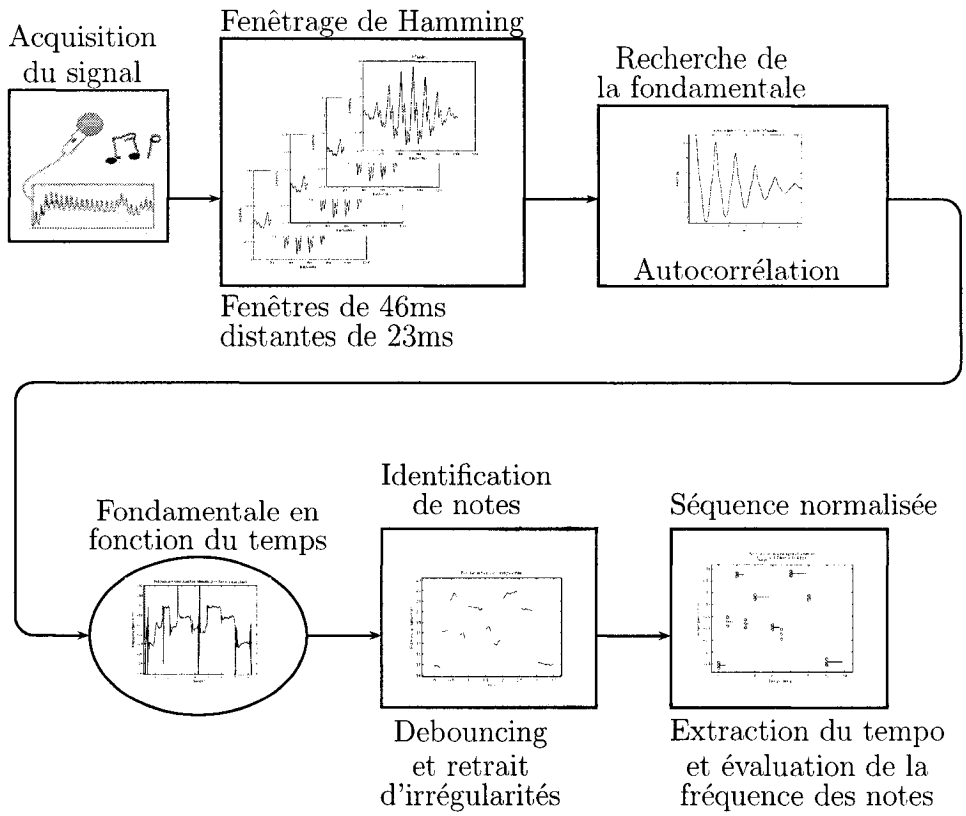

FIG. 2.1 Schéma bloc du prétraitement du son.

2.1 Découpage d'un extrait sonore

Une fois l'extrait sonore $s(t)$ de durée T recueilli à la fréquence d'échantillonnage f_{ech} avec un total de $S = f_{ech}T$ échantillons, on recherche ensuite la représentation de la fréquence fondamentale en fonction du temps $f(t)$. Pour ce faire, on analysera des fenêtres de courte durée. Ces fenêtres seront des découpages du signal original pris à intervalles réguliers, soit à la fréquence de découpage $f_{dec} = 43Hz$.

On choisit des fenêtres de taille N échantillons pour évaluer la fréquence fondamentale. Plus N est grand, plus on peut évaluer les basses fréquences. Avoir des

tranches de taille $N = 1024$ et un taux d'échantillonnage $f_{ech} = 22.05kHz$ permet d'identifier convenablement les basses fréquences jusqu'à $55Hz$. La figure 2.2 illustre la procédure de découpage. Les fenêtres de découpage sont de taille N échantillons et sont distancées de $M = N/2$ échantillons.

La fréquence de découpage $f_{dec} = \frac{f_{ech}}{M}$ ne doit pas être trop basse, sinon on ne pourrait plus distinguer les différentes notes d'un air chanté. On observe qu'un locuteur peut chanter des notes aussi courte que $125ms$. Une bonne fréquence de découpage est alors de l'ordre de 30 à $50Hz$.

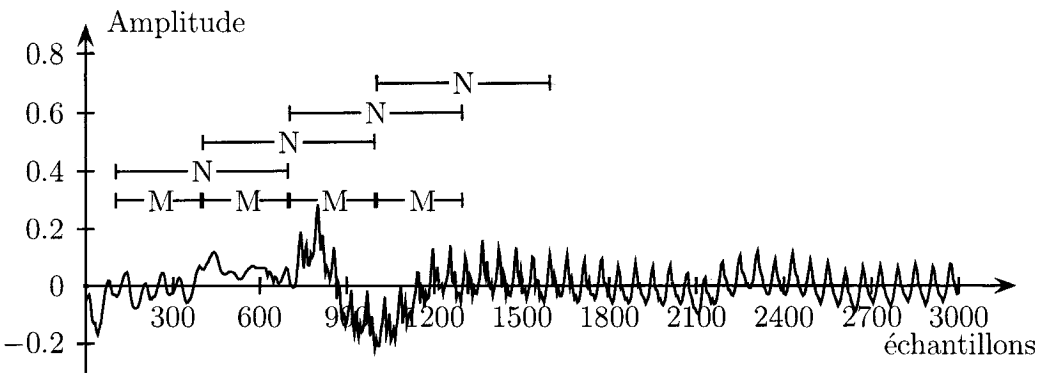

FIG. 2.2 Découpage d'un extrait musical

Voyons maintenant la démarche de découpage plus en détail. On dispose au départ d'un signal sonore, $s(t)$, recueilli à l'aide d'un microphone. Ce signal possède un nombre total de S échantillons recueillis à la fréquence f_{ech} et se représente comme suit :

$$s(t), t \in \{0, 1, ..., S\} \tag{2.1}$$

On décrit par la suite le signal dans la l^e fenêtre de découpage.

$$x_l(n) = s(Ml + n), l \in \{0, 1, ..., L - 1\} \; et \; n \in \{0, 1, ..., N - 1\} \tag{2.2}$$

Où L est le nombre total de fenêtres de découpage et est défini comme :

$$L = \left\lfloor \frac{S - (N - M)}{M} \right\rfloor \tag{2.3}$$

Sur chaque fenêtre de découpage $x_l(n)$, on ajoute la fenêtre de Hamming. Cette dernière permet de minimiser les harmoniques résiduelles du sinus cardinal provenant de la transformée de Fourier de la fenêtre carrée de découpage. On analyse donc plutôt le signal suivant :

$$\tilde{x}_l(n) = x_l(n)w(n) \tag{2.4}$$

où $w(n)$ est la fenêtre de Hamming :

$$w(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) \tag{2.5}$$

2.1.1 Incertitude des fenêtres

Sachant que l'écart type de la fenêtre de Hamming de largeur 1 est de $\sigma_{Ham} = 0.201$, il est possible de déterminer le Δt de la fenêtre de découpage sur laquelle la fenêtre de Hamming s'applique :

$$\Delta t = 2\frac{N\sigma_{Ham}}{f_{ech}} = 2\frac{2048 \cdot 0.201}{22.05kHz} = 37.3ms \tag{2.6}$$

D'où l'on a déduit que la fréquence minimale pouvant être observée est de $f_{min} = \frac{1}{\Delta t} = 26.8Hz$. Ceci permet également de définir l'écart type des pics de partiel observés dans le spectre (domaine spectral), qui est directement lié à l'écart type de la fenêtre de découpage selon la relation :

$$\Delta f \Delta t = \frac{1}{2} \tag{2.7}$$

Cette relation est utile lorsque l'on tient à connaître l'incertitude de la fréquence fondamentale évaluée dans le spectre des fréquences à partir des fenêtres de découpage.

2.1.2 Notation matricielle d'un extrait sonore

Pour effectuer l'analyse d'un extrait sonore en fonction du temps, il est plus pratique de réorganiser les fenêtres à analyser dans une matrice X contenant toutes les fenêtres de Hamming. On a donc les vecteurs $\vec{x}_l = (\tilde{x}(0)\ \tilde{x}(1)\ ...\ \tilde{x}(N-1))^T$ représentant les fenêtres de découpage auxquelles ont été appliquées les fenêtres de Hamming permettant de poser la représentation matricielle de l'extrait sonore émis par le locuteur :

$$X = \left(\begin{array}{ccccccc} \vec{x}_0 & \vec{x}_1 & ... & \vec{x}_l & ... & \vec{x}_{L-1} \end{array} \right) \tag{2.8}$$

Dans cette matrice, chaque colonne correspond à la l^e fenêtre de découpage de l'extrait sonore. La distance entre ces fenêtres est de $\Delta t = \frac{M}{f_{ech}} = \frac{512}{22.05kHz} = 23.2ms$.

2.2 Extraction de la fréquence fondamentale

Une fois la séquence musicale découpée en fenêtres caractérisant l'évolution du signal dans le temps, on s'intéresse ensuite à l'identification des expressions qui la composent, c'est-à-dire les notes de musique. Pour ce faire, nous utiliserons comme exemple la séquence de la figure 2.3.

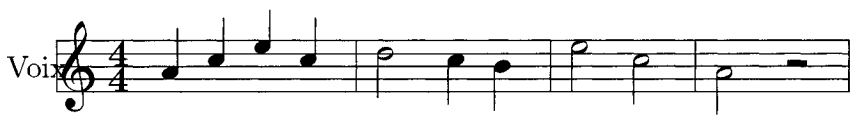

FIG. 2.3 Séquence musicale

En ce qui concerne la voix humaine, lorsque l'on chantonne, fredonne ou siffle un air musical, on peut considérer que la région spectrale d'intérêt demeure à l'intérieur de $la_0 = 55Hz$ à $la_5 = 1.76kHz$, comme nous l'avons vu au chapitre 1.

2.2.1 Autocorrélation dans le domaine temporel

Une première méthode pour évaluer la fréquence fondamentale à un instant donné correspondant à une fenêtre de découpage du signal original est d'effectuer une simple autocorrélation de cette fenêtre (Rabiner et Juang, 1993). Cela permet d'extraire la période du signal qui est l'inverse de la fréquence fondamentale. En effet, l'emplacement du plus grand pic d'autocorrélation correspond à la période du signal de la fenêtre de découpage.

On définit l'autocorrélation $a(\tau)$ d'un signal $x(n)$ de longueur N comme étant :

$$a(\tau) = \sum_{n=0}^{N-\tau-1} x(n)x(n+\tau) \tag{2.9}$$

Par exemple, pour un air chanté, l'autocorrélation de la fenêtre de découpage de la figure 2.4 donnerait le signal de la figure 2.5.

Selon cette figure d'autocorrélation (figure 2.5), le signal aurait une période d'environ $5.85ms$. La période du signal est liée à sa fréquence fondamentale selon la relation :

$$T = 1/f \tag{2.10}$$

On obtient pour le signal de la figure 2.4 une fréquence de $170.9Hz$, ce qui se trouve dans l'intervalle $[mi_1^\flat, mi_1^\natural]$.

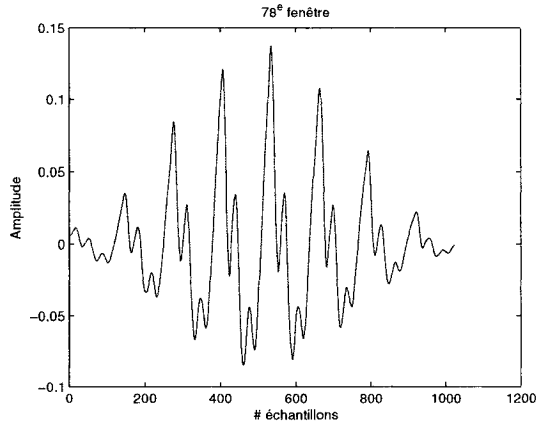

FIG. 2.4 Fenêtre de découpage d'un air chanté.

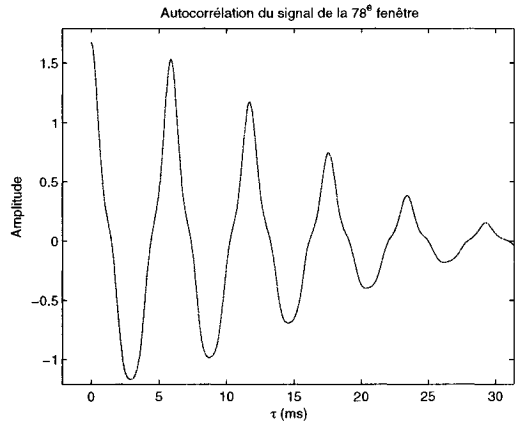

FIG. 2.5 Autocorrélation de la fenêtre de découpage de la figure 2.4.

Une fois que le maximum local d'autocorrélation associé à la période du signal a été déterminé, il est possible de l'évaluer avec davantage de précision. Pour ce faire, on procède à une interpolation de la courbe d'autocorrélation par une courbe du second degré, c'est-à-dire que l'on prend le sommet de la parabole passant par le maximum trouvé, $a(\tau_{max})$, et ses voisins, $a(\tau_{max} \pm 1)$.

Cette méthode s'avère assez efficace à basse fréquence pour évaluer la tonalité avec précision, mais lorsque l'on passe dans le domaine des hautes fréquences tel qu'un

air chanté par une locutrice ou un air sifflé, l'incertitude quant à l'évaluation du maximum local devient trop importante.

Cette incertitude découle de la longueur des pas d'échantillonnage $\Delta t_{ech} = 1/f_{ech} = 22.7\mu s$. Plus la période du signal est petite, plus l'incertitude relative quant à la période du signal sera grande. On doit donc s'assurer que cette incertitude ne dépasse pas la moitié d'un demi-ton. On peut poser la période minimale d'un signal comme étant :

$$T_{min} = \frac{\Delta t_{ech}}{2^{\frac{0.5}{12}} - 1} = \frac{22.7\mu s}{0.0293} = 775\mu s \tag{2.11}$$

La valeur $2^{\frac{0.5}{12}} - 1$ représente la variation d'environ 3 % que l'on retrouve dans une variation de fréquence de la moitié d'un demi-ton.

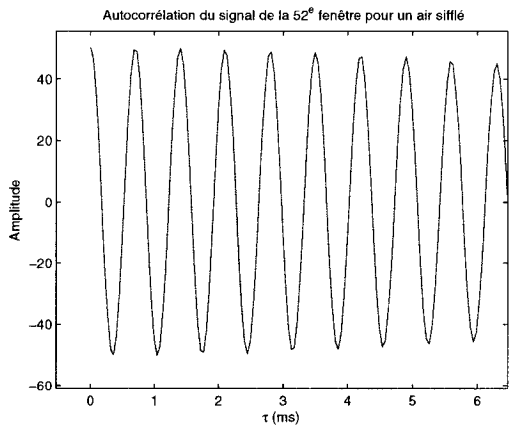

FIG. 2.6 Autocorrélation d'une fenêtre de découpage pour un air sifflé.

L'équation 2.11 permet donc de déduire la fréquence maximale pour laquelle l'autocorrélation sera bonne, soit : $f_{max} = \frac{1}{T_{min}} = 1.29kHz$. Cette limite n'est toutefois pas suffisante pour couvrir la plage d'intérêt $[la_0, la_5]$.

Pour résoudre ce problème, nous ferons d'abord une première évaluation de la fréquence fondamentale. Si l'on dépasse f_{max}, nous utiliserons ensuite le second sommet de l'autocorrélation pour évaluer la période du signal avec plus de préci-

sion[2]. Cette situation se présente, par exemple, dans le cas d'une séquence musicale sifflée (voir figure 2.6).

2.2.2 Sonogramme

La transformée de Fourier, appliquée à la matrice des fenêtres X, vue à l'équation 2.8, selon la première dimension (verticalement à chacun des vecteurs de fenêtre), donne le sonogramme de la séquence musicale. Un sonogramme permet de voir l'évolution du spectre dans le temps (voir figures 2.7 et 2.8).

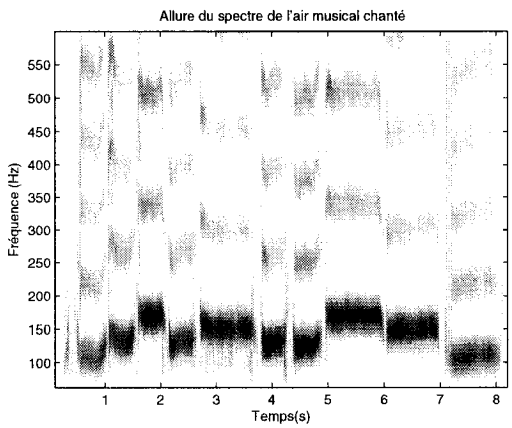

FIG. 2.7 Sonogramme de la séquence musicale chantée de la figure 2.3.

Dans la séquence sifflée (figure 2.7), le spectre sonore est relativement monochromatique, c'est-à-dire que l'on observe une seule harmonique spectrale : la fondamentale. Cependant, dans la séquence chantée (figure 2.7), qui contient davantage d'harmoniques spectrales, on observe que certaines notes partagent des harmoniques spectrales. Par exemple, la première et la 3^e note partagent l'harmonique $f \approx 330Hz$. On y reviendra dans le chapitre 3 traitant de la représentation des intervalles mélodiques.

[2]Dans ce cas, f_{max} devient $f_{max} = \frac{2}{2T_{min}} = \frac{2}{775\mu s} = 2.58kHz$.

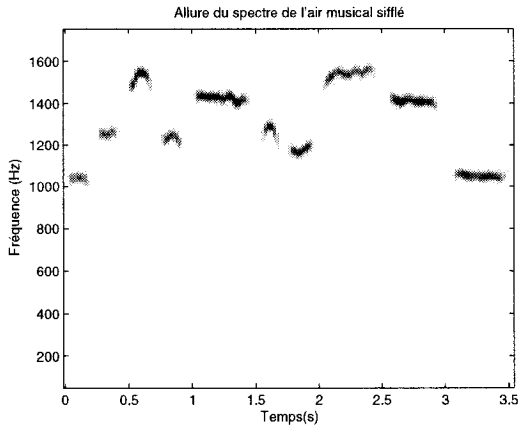

FIG. 2.8 Sonogramme de la séquence musicale sifflée de la figure 2.3.

2.2.3 Fréquence fondamentale en fonction du temps

On obtient, après avoir analysé chacune des fenêtres de découpage à l'aide de la méthode de l'autocorrélation, $f(t)$ la représentation de la fréquence fondamentale en fonction du temps. On peut observer, aux figures 2.9 et 2.10, la fréquence fondamentale $f(t)$ des séquences musicales représentées dans les sonogrammes des figures 2.7 et 2.8.

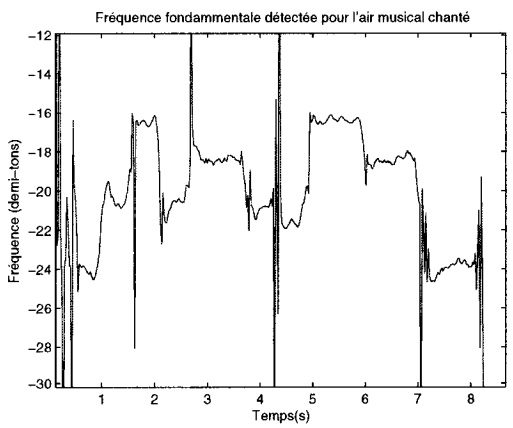

FIG. 2.9 Fréquence fondamentale détectée pour la séquence musicale chantée de la figure 2.3. La fréquence est donnée en demi-tons avec 0 pour le $la_3 = 440Hz$.

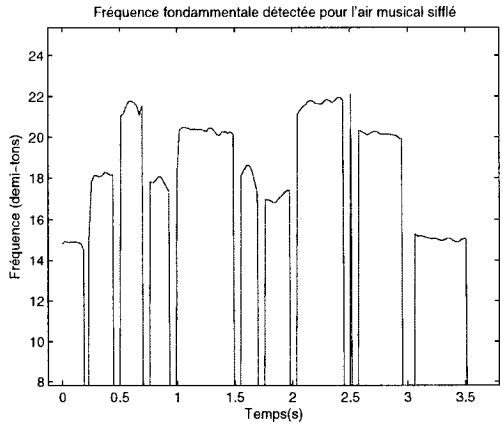

FIG. 2.10 Fréquence fondamentale détectée pour la séquence musicale sifflée de la figure 2.3.

2.3 Extraction des notes

Après l'extraction de la fréquence fondamentale $f(t)$, on cherche à identifier clairement les notes musicales. Pour ce faire, on définit le signal binaire de présence de notes $\nu(t)$. Ce dernier prend la valeur 1 lorsqu'il y a présence d'une note et la valeur 0 en l'absence d'une note ou entre deux notes.

Pour initialiser ce signal $\nu(t)$, il est raisonnable de lui attribuer la valeur 1 quand l'amplitude du son est significative et quand la fréquence fondamentale détectée ne varie pas trop fortement. Autrement, sa valeur est de 0. On note l'amplitude du signal en fonction du temps comme étant $A(t)$. Le signal $\nu(t)$ se décrit donc comme étant :

$$\nu(t) = \begin{cases} 1 & \text{si } A(t) > A_{min} \text{ et } \frac{\partial f(t)}{\partial t} < \gamma \\ 0 & \text{sinon} \end{cases} \tag{2.12}$$

On utilise ici A_{min} comme étant l'amplitude minimale devant être émise pour détecter une note. Une bonne valeur de A_{min} se situe environ à $1,5$ % de l'amplitude maximale pouvant être échantillonnée. Le seuil γ est limité à une variation de 0,5 demi-ton entre deux fenêtres distantes de 23 ms.

2.3.1 Détection de plateaux

Cependant, il s'est avéré expérimentalement que cette simple segmentation n'était pas suffisante. On a donc ajouté un autre critère basé sur la variance de la fréquence fondamentale à l'intérieur d'une région temporelle. Ainsi, si on observe que la fréquence fondamentale ne varie pas plus que Δf_{plt} (plt pour plateau) dans une région temporelle, on peut associer une note au plateau formé (voir figure 2.11). On segmente alors les différents plateaux trouvés en posant $\nu(t) = 0$ entre ceux-ci. Une bonne valeur de Δf_{plt} se situe environ à un demi-ton.

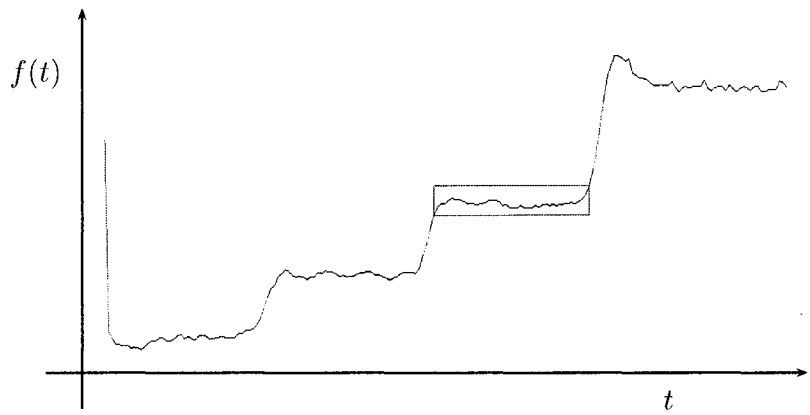

FIG. 2.11 Détection de plateaux où la fréquence varie de moins de Δf_{plt}=1 demi-ton.

2.3.2 Anti-rebond

Finalement, on s'assure que le signal binaire de présence de notes $\nu(t)$ ne varie pas trop rapidement. On éliminera alors les notes de trop courte durée ayant été détectées. Ces dernières sont une des principales sources d'erreurs lors de la reconnaissance de séquences musicales.

Dans notre cas, on éliminera les notes ayant une durée de moins de $\Delta t_{min} = 125ms$. Cela donne un étalement en nombre de fenêtres de découpage de $125ms * f_{dec} \simeq 3$ fenêtres.

Un exemple de signal $\nu(t)$ est donné à la figure 2.12b. Ce signal permet de ne garder que les régions de la fréquence fondamentale associées aux notes chantées (voir figure 2.12c).

2.4 Attaques de note et extraction du tempo

Le signal de présence de notes ($\nu(t)$) permet également d'identifier les attaques de notes (fronts montants du signal $\nu(t)$) et leur durée (période entre deux fronts montants).

Une fois que l'on a détaché les notes du signal $f(t)$, grâce à la fonction de présence de notes $\nu(t)$, on cherche à évaluer le tempo[3] de la séquence musicale. On s'attend à ce que ces notes, d'une durée variable, respectent un certain tempo. On cherchera d'abord la plus petite période commune séparant les diverses attaques de note. Cette période est considérée comme étant un temps et on suppose que les durées des différentes notes émises sont des multiples entiers de cette plus petite période. Pour ce faire, on utilise un nouveau signal nommé $O(t)$[4] représentant les attaques de note en fonction du temps et de leur incertitude. Ce signal est constitué de gaussiennes pour chacune des attaques de note (voir figure 2.12d). On choisie pour ces gaussiennes un écart-type du quart de la plus petite distance entre deux notes. Ceci évite d'avoir un recouvrement entre les gaussiennes et permet d'identifier aisément le tempo à l'aide de l'autocorrélation du signal $O(t)$.

[3]Le tempo est l'allure (rapidité relative) d'exécution d'un morceau de musique. Ce dernier peut s'exprimer comme une fréquence. Par exemple, 90 noires à la minute (90 bpm).

[4]Ici, on emploie O pour "onset".

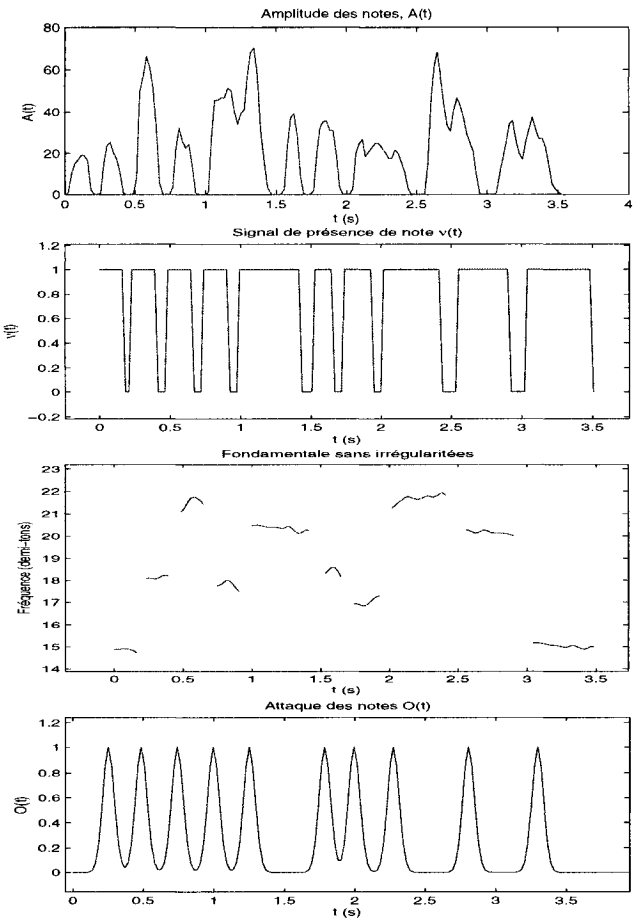

FIG. 2.12 a) Amplitude des notes $A(t)$. b) Signal de présence de notes $\nu(t)$. c) Fréquence fondamentale détectée sans irrégularités. d) Signal d'attaques de note $O(t)$.

Une fois le tempo déterminé, il est aisé de décrire l'air donné par l'usager en une suite de notes dont la durée est représentée par de petits entiers (voir figures 2.13 et 2.14). La séquence mélodique s'écrit donc comme une suite de paires fréquence-durée associées aux notes.

Dans ces figures, on donne également la variance de la note chantée. Une grande variance signifie habituellement une note ayant été mal attaquée et pour laquelle l'usager tente de réajuster son tir (mauvais chanteur). Il peut également s'agir d'une voix en vibrato.

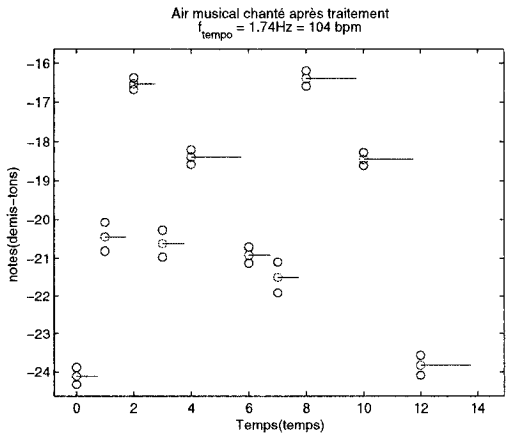

FIG. 2.13 Séquence de notes déterminée après le traitement de la séquence musicale chantée de la figure 2.3.

2.5 Intervalles mélodiques et temporels

Comme nous l'avons vu à la section 1.2, il est préférable de considérer les intervalles mélodiques plutôt que les notes. De même, pour les durées de notes, on utilisera les ratios de durées de notes, dénotés IOIr[5] (Pardo et al., 2002). Ceci permet d'avoir un encodage invariant selon les transpositions et le tempo. La représentation de la mélodie devient donc une séquence de paires d'intervalles mélodiques et d'IOIr. On donne, à la figure 2.15, l'exemple d'une séquence mélodique transformée en séquence d'intervalles mélodiques et de ratios d'intervalles temporels.

2.6 Conclusion du chapitre

Dans ce chapitre, nous avons vu comment il est possible à partir de données brutes recueillies par un locuteur d'extraire des notes musicales. De ces séquences de notes, on ne retient finalement que les intervalles mélodiques (IM) et les ratios d'intervalles temporels (IOIr). Mais avant d'analyser la séquence musicale, nous allons nous

[5]Inter onset interval ratio (IOIr).

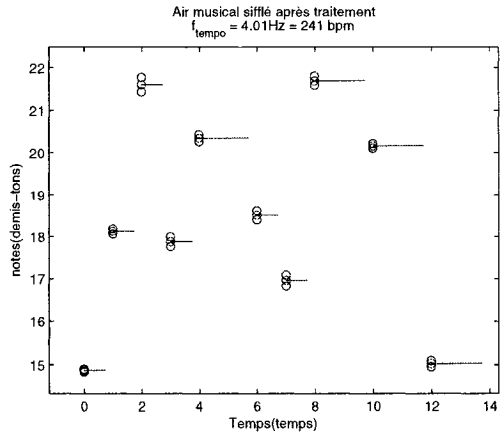

FIG. 2.14 Séquence de notes déterminée après le traitement de la séquence musicale sifflée de la figure 2.3.

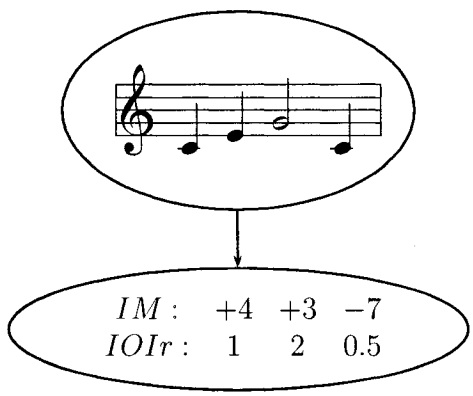

FIG. 2.15 Extraction des intervalles mélodiques (distance entre deux notes), IM, et des ratios d'intervalles temporels, IOIr, à partir de la séquence de notes obtenue.

attarder à la rendre plus respectueuse de son aspect mélodique. Pour ce faire, nous modifierons les intervalles mélodiques représentés par de simples scalaires (4, 3, et -7 demi-tons) dans l'exemple de la figure 2.15 pour les transposer dans un espace de plus haute dimension leur permettant de se placer les uns par rapport aux autres afin de respecter leurs « harmonicités » mutuelles.

CHAPITRE 3

REPRÉSENTATION DES INTERVALLES MÉLODIQUES

À l'écoute de notes musicales, on ressent intuitivement l'harmonie produite par deux sons. On entend par harmonie le fait que ces sons s'agencent bien ensemble. Par exemple, les notes do et sol présentent une bonne harmonie, alors que do et ré présentent une harmonie moins forte. L'objectif de ce chapitre est de trouver une manière d'interpréter les intervalles mélodiques[1] afin de tenir compte de l'harmonie. L'harmonie d'une chanson est en quelque sorte le cadre d'une chanson. Si les notes modifiées demeurent dans ce cadre, la chanson devrait quand même être reconnue. Pour tenir compte de l'harmonie, on utilisera la « corrélation harmonique ». Cette dernière se définit comme la ressemblance auditive entre deux notes. Nous verrons qu'elle dépend directement de la distance entre le spectre de la première note et celui de la seconde.

Cette notion de corrélation harmonique est importante, car l'identification d'un air musical repose surtout sur l'harmonie relative de la suite des intervalles mélodiques. Il est donc plus astucieux de définir la proximité de deux notes (corrélation harmonique) en se basant sur la similitude de leur spectre que sur la proximité de leur fréquence fondamentale.

[1]Comme nous l'avons vu au chapitre 1, l'intervalle mélodique se définit comme la différence entre deux notes consécutives (voir figure 1.4).

3.1 Corrélation harmonique

Lorsque l'on chante le $la_3 = 440Hz$ ou le $mi_3 = 660Hz$, les fréquences $440Hz$ et $660Hz$ ne sont pas les seules à être émises. Un grand nombre d'harmoniques, associées à ces notes, sont également générées (voir section 1.1.2). Un grand nombre de ces harmoniques sont communes aux deux notes, telles que $1320Hz$, $2.64kHz$, $3.96kHz$, etc. En fait, puisque l'on a $f_2 = \frac{3f_1}{2}$, les harmoniques de fréquences $f_i = 3if_1 = 2if_2,\ i \in \mathbb{N}$ sont communes aux spectres des deux notes.

Dans cet exemple, les deux notes : la et mi possèdent un rapport de fréquence de $\frac{3}{2}$, ce qui donne 7 demi-tons[2] ($2^{7/12} = 1,498 \simeq \frac{3}{2}$). Cet écart de fréquence se définit en musique comme une quinte (voir section 1.2). Les notes ayant une distance de 7 demi-tons sont donc fortement corrélées (voir figure 3.2).

De la même manière, les intervalles mélodiques (utilisés pour décrire les séquences) distants de 7 demi-tons sont fortement corrélés. Par exemple, si l'on chante dans un extrait $[mi, do]$ ($r = -4$) au lieu de $[mi, sol]$ ($r = +3$) (voir figure 3.3a), on a un $\Delta r = 7$ et la mélodie est respectée puisque les notes do et sol sont fortement corrélées.

3.1.1 Définition de la corrélation harmonique

Pour définir la corrélation harmonique, on considère le nombre moyen d'harmoniques spectrales partagées entre deux notes par rapport à leur nombre total d'harmoniques spectrales. On aura alors pour deux notes :

- identiques (unisson) : corrélation de 1 (les deux notes partagent l'ensemble de leurs harmoniques spectrales)

[2] $\frac{f_{mi}}{f_{la}} = \frac{660Hz}{440Hz} = 3/2$ et $r_{mi} - r_{la} = 7$.

- distantes d'un octave : corrélation de $\frac{3}{4}$ (une des deux notes partage l'ensemble de ses harmoniques spectrales et la seconde en partage la moitié)
- distantes d'une quinte : corrélation de $\frac{5}{12}$ (rapport de fréquence de $\frac{3}{2}$, donc $\frac{1}{2}$ et $\frac{1}{3}$ des harmoniques en commun)
- distantes d'une quarte : corrélation de $\frac{7}{24}$ (rapport de fréquence de $\frac{4}{3}$)

et ainsi de suite. Il est toutefois préférable de définir une fonction continue donnant la corrélation harmonique de tous les intervalles. En effet, les intervalles ne sont pas toujours associables à des ratios rationnels facilement identifiables.

Nous allons donc rechercher un moyen de quantifier la corrélation $H(r)$ de deux notes distantes d'un intervalle $r \in \mathbb{R}$. Une corrélation de 0 signifie que les deux notes de l'intervalle ne présentent aucune harmonie. Une corrélation de 1 pour sa part serait un cas où les deux notes sont identiques. Pour déterminer la fonction de corrélation harmonique $H(r)$, on considérera le niveau de corrélation entre deux spectres idéaux de deux notes ayant pour fréquences fondamentales f_0 et Rf_0. Ces notes possèdent donc un rapport de fréquence de $\frac{Rf_0}{f_0} = R$. Le ratio de fréquence est lié à l'intervalle r selon :

$$R = 2^{r/D} \tag{3.1}$$

Chacun des spectres $z(f, f_{fond})$ de note de fréquence fondamentale f_{fond} est représenté par le produit de convolution (noté $\odot$) d'un peigne de dirac (noté $comb(x)$[3]), représentant les harmoniques d'une note, et une gaussienne, représentant l'étalement spectral de chacune des harmoniques :

$$z(f, f_{fond}) = e^{\frac{-f^2}{\sigma^2}} \odot comb\left(\frac{f}{f_{fond}}\right) \tag{3.2}$$

[3] $comb(x) = \sum_{i=1}^{\infty} \delta(x - i)$, ici le peigne commence à 1 pour la première harmonique (fondamentale).

La corrélation entre les spectres de deux notes, dont le rapport de fréquence est R, permet de définir la corrélation harmonique :

$$h_1(R) = C \int_0^\infty \left(e^{\frac{-f^2}{\sigma^2}} \odot comb\left(\frac{f}{f_0}\right)\right) \cdot \left(e^{\frac{-f^2}{\sigma^2}} \odot comb\left(\frac{f}{Rf_0}\right)\right) df \tag{3.3}$$

C est une constante de normalisation.

Au lieu du peigne de dirac convolué avec une gaussienne, on emploiera une fonction sinusoïdale à la puissance $2d$: $cos^{2d}(x)$. Cette fonction $cos^{2d}(x)$ est une approximation du peigne convolué avec la gaussienne et simplifie l'évaluation d'une corrélation tel que celle donnée à l'équation 3.3. Ici, $d \in \mathbb{N}$ est un paramètre à ajuster pour obtenir une corrélation raisonnable (présence de pics forts pour l'octave, $\frac{3}{4}$, la quinte, $\frac{5}{12}$, etc.) (voir figures 3.1 et 3.2).

$$z(f, f_{fond}) = e^{\frac{-f^2}{\sigma^2}} \odot comb\left(\frac{f}{f_{fond}}\right) \Rightarrow \cos^{2d}\left(\frac{\pi f}{f_{fond}}\right) \tag{3.4}$$

L'équation 3.3 est alors approximé par :

$$h_2(R) = C \int_0^\infty \cos^{2d}\left(\frac{\pi f}{f_0}\right) \cdot \cos^{2d}\left(\frac{\pi f}{Rf_0}\right) df$$

Ensuite, afin d'accélérer les calculs, on effectuera une sommation à chacun des pics d'harmonique $f = kf_0$ du son de référence de la fréquence $f_{fond} = f_0$. On aura alors une corrélation harmonique de la forme :

$$\begin{aligned} h_3(R) &= C \sum_{k=1}^{\infty} \cos^{2d}\left(\frac{\pi f}{f_0}\right)\Big|_{f=kf_0} \cdot \cos^{2d}\left(\frac{\pi f}{Rf_0}\right)\Big|_{f=kf_0} \\ &= C \sum_{k=1}^{\infty} 1 \cdot \cos^{2d}\left(\frac{\pi k}{R}\right) \end{aligned} \tag{3.5}$$

Finalement, nous prendrons en considération que nous donnerons davantage d'importance aux premières harmoniques k. Ceci évite de tenir compte du bruit provenant des harmoniques spectrales de hautes fréquences lorsque k est élevé. Pour ce faire, on utilise une fonction en forme de cloche centrée sur zéro, les premières harmoniques contribuant ainsi davantage à la perception de la corrélation harmonique. Pour ce faire, on ajoutera à l'équation 3.5 une lorentzienne centrée sur $f = 0Hz$ avec une largeur de σ_h à mi-hauteur. Cette variable σ_h sera un autre paramètre à ajuster pour observer les pics désirés dans la fonction de corrélation harmonique.

$$h_4(R) = C \sum_{k=1}^{\infty} \frac{\cos^{2d}\left(\frac{\pi k}{R}\right)}{1 + \frac{k^2}{\sigma_h^2}} \tag{3.6}$$

En utilisant une échelle logarithmique pour le rapport de fréquence telle que l'échelle diatonique, on obtient ce que l'on définira comme la « corrélation harmonique ». On substitue le rapport de fréquence R par l'intervalle équivalent r, les deux étant liés par la relation 3.1.

$$H_{tmp}(r) = C \sum_{k=1}^{\infty} \frac{\cos^{2d}\left(\pi k 2^{\frac{-r}{D}}\right)}{1 + \frac{k^2}{\sigma_h^2}} \tag{3.7}$$

Dans le cas de l'échelle chromatique, on a $D = 12$ et r représente l'intervalle en demi-tons. Pour avoir une meilleure idée de cette fonction, on présente, à la figure 3.1a en ordonnée, les harmoniques communes entre une note de fréquence $f_1 = Rf_0$ et une note de référence de fréquence f_0. En abscisse, les différents intervalles sont présentés en demi-tons. Comme on peut le remarquer dans le cas de l'unisson ($r = 0$), toutes les harmoniques sont partagées. À l'octave supérieur ($r = 12$), on constate qu'une harmonique sur deux est partagée. Par contre, à l'octave inférieur, toutes les harmoniques sont partagées, car $f_1 = f_0/2$, ce qui fait en sorte que les harmoniques de f_0 sont nécessairement des harmoniques de f_1.

a)

b)

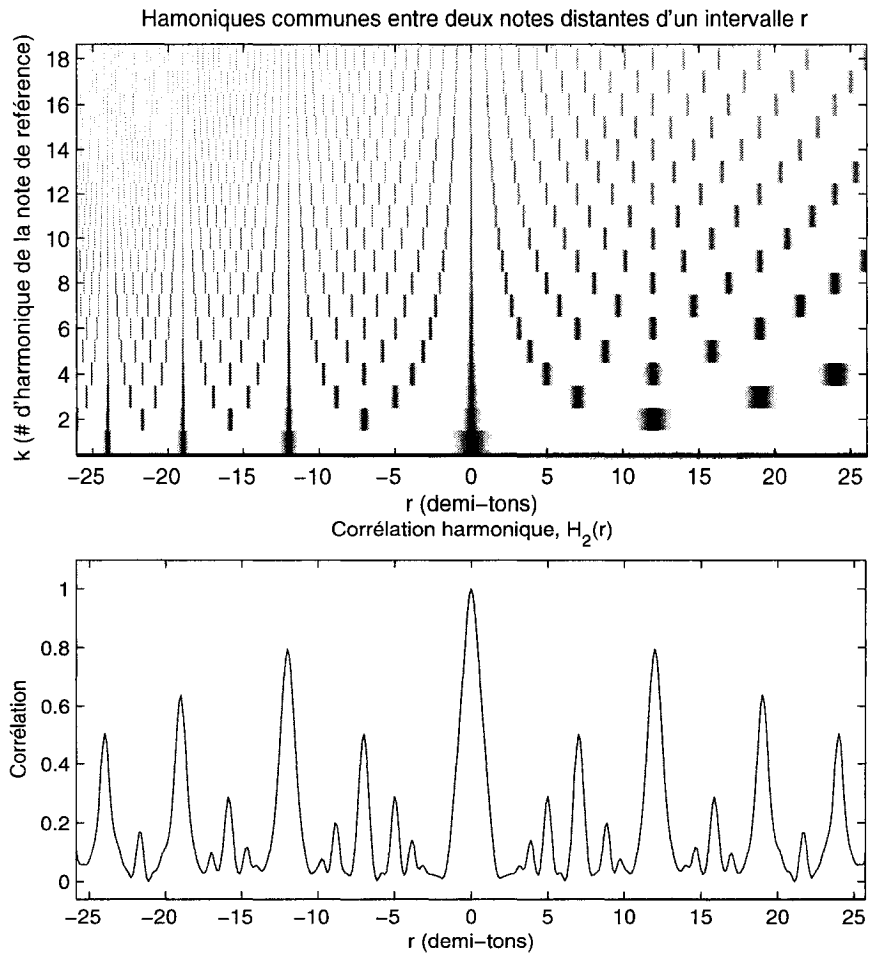

FIG. 3.1 a) Représentation en ordonnée des harmoniques communes entre deux sons distancés d'un intervalle de r demi-tons (en abscisse). b) Fonction de la corrélation harmonique $H(r)$, moyenne des harmoniques communes.

La fonction $H_{tmp}(r)$ n'est donc pas symétrique, car on n'a pas considéré la moyenne des harmoniques communes, mais seulement les harmoniques de la première note qui sont communes avec celles de la seconde note. Pour remédier à ce problème, on n'a qu'à effectuer la moyenne de $H_{tmp}(r)$ et de $H_{tmp}(-r)$. Ainsi, une quinte ascendante présentera la même corrélation qu'une quinte descendante par rapport à l'unisson.

$$H(r) = \frac{H_{tmp}(r) + H_{tmp}(-r)}{2} \tag{3.8}$$

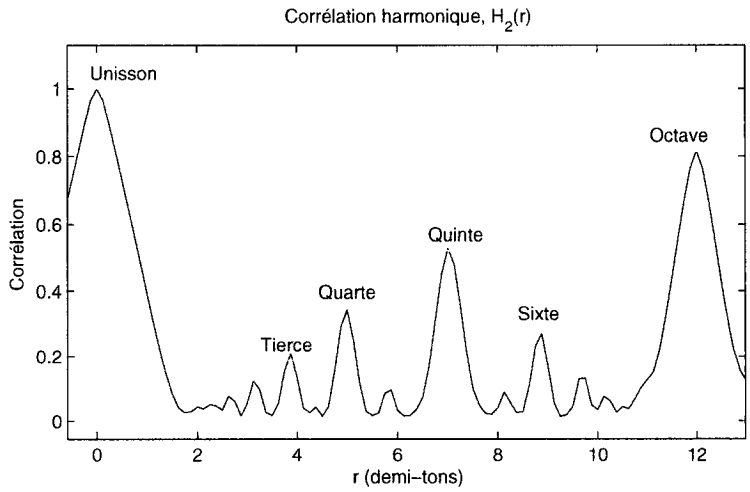

FIG. 3.2 Corrélation harmonique pour les 12 premiers demi-tons, $H(r)$.

Cette fonction de corrélation harmonique est donnée aux figures 3.1b et 3.2. On peut y observer les intervalles importants en musique tels que la quinte, la quarte, la sixte, etc. On a ajusté les différents paramètres (d, σ_h) de la fonction afin d'obtenir des corrélations raisonnables pour les différents intervalles mélodiques ($\frac{3}{4}$ pour l'octave, $\frac{5}{12}$ pour la quinte, etc.).

Cette définition de la corrélation harmonique sera pratique pour définir la distance relative entre les divers intervalles mélodiques. Par exemple, une quarte ascendante est assez proche d'une seconde descendante (différence d'une quinte entre les deux), et ce, même si la seconde descendante possède une faible corrélation avec l'unisson, tandis que la quarte est fort corrélée avec l'unisson.

3.2 Utilité de la corrélation harmonique

L'utilisation de la corrélation harmonique pour définir la similitude de différents intervalles mélodiques permet de respecter l'harmonie que pourraient présenter deux séquences semblables. Il pourrait arriver par exemple à un chanteur de vouloir

reproduire une séquence musicale dans laquelle il ne se souviendrait pas de l'air exact, mais plutôt d'une variante de cet air. Un chanteur pourrait bien aussi être limité dans son jeu de notes et ne pas pouvoir chanter des notes très aiguës ou très graves. Il serait alors contraint à chanter les notes de l'octave supérieur ou inférieur. Ces situations sont illustrées à la figure 3.3. Les notes superposées représentent des variantes modifiant peu l'harmonie de la séquence.

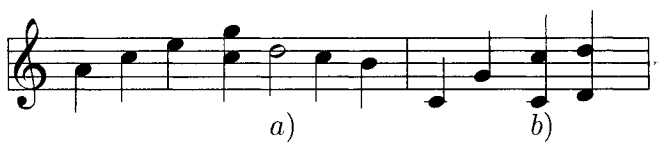

FIG. 3.3 a) Variantes mélodiques d'une séquence musicale. b) Séquence musicale chantée dans un registre limité (notes graves).

Dans notre système de reconnaissance de mélodies musicales, nous avons tenu compte de cette corrélation harmonique lorsque l'on cherche à identifier une mélodie à l'aide des intervalles mélodiques (voir section 2.5). Ceci permet, comme on peut l'observer à la figure 3.4 de trouver une mélodie chantée dans un registre limité. Ici, le chanteur ne pouvant pas chanter les notes aiguës de la chanson « The Entertainer », s'est limité aux notes graves.

3.3 Recherche d'une représentation harmonique des intervalles

Maintenant que l'on a défini la corrélation harmonique, on peut rechercher une manière de représenter les intervalles mélodiques qui respecterait cette corrélation. Utiliser de simples scalaires (0 pour l'unisson, 5 pour la quarte, 7 pour la quinte, etc.) ne permet pas de fournir l'information sur les corrélations harmoniques que peuvent présenter les différents intervalles mélodiques entre eux. Nous allons donc transposer les intervalles mélodiques dans un espace de plus haute dimension, ce

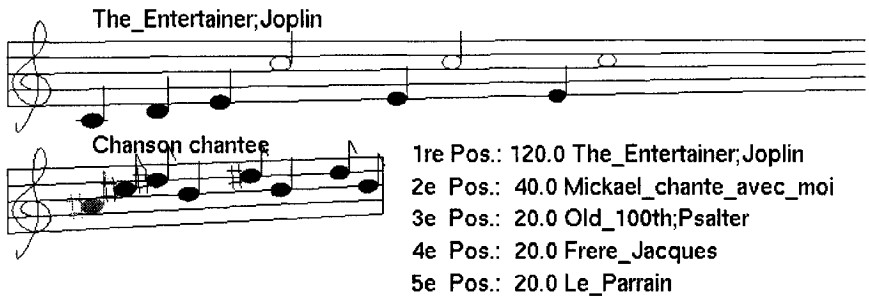

FIG. 3.4 Résultat d'une recherche à l'aide du programme de reconnaissance de séquences musicales dans le cas d'une variante mélodique. On retrouve, dans le coin inférieur droit, les chansons ayant obtenu les meilleurs scores de ressemblance avec la mélodie chantée.

qui leur permettra de se placer les uns par rapport aux autres afin de respecter la corrélation harmonique.

Pour ce faire, nous associerons à chaque intervalle mélodique $r = ..., -2, -1, 0, 1, ...$ demi-tons un vecteur $\vec{u}_r$ dans un espace en N dimensions[4]. On pourra alors positionner dans cet espace les intervalles mélodiques de manière à ce qu'ils respectent la corrélation harmonique.

3.3.1 Représentation matricielle de l'ensemble des intervalles mélodiques

On représente l'ensemble des vecteurs d'intervalle mélodique $\vec{u}_r$ (unisson, seconde, tierce, etc.) par une matrice $U_{N \times M}$ contenant M vecteurs d'intervalle mélodique

[4]La dimensionnalité N de l'espace des vecteurs d'intervalle mélodique est un paramètre à ajuster. Sa valeur optimale sera expliquée dans la section 3.4

ou VIM en N dimensions.

$$U = (\underset{}{\vec{u}_{-(M-1)/2}} \ \ldots \ \underset{Unisson}{\vec{u}_0} \ \ldots \ \underset{5^{te}asc.}{\vec{u}_7} \ \ldots \ \vec{u}_{(M-1)/2}) \tag{3.9}$$

Typiquement, on a $M = n^{bre}\ d'octaves \times D + 1$ intervalles mélodiques où D est le nombre de subdivisions par octave ($D = 12$ selon l'échelle chromatique). De cette manière, si l'on désire couvrir deux octaves dans l'échelle chromatique, il faut $M = 25$ vecteurs représentant les 25 intervalles à couvrir, soit de l'intervalle $r = -12$ demi-tons à $r = 12$ demi-tons, en passant par $r = 0$ pour l'unisson.

Le but recherché est que chacun de ces vecteurs d'intervalle mélodique respecte la corrélation harmonique qu'il doit avoir avec les autres vecteurs d'intervalle mélodique.

3.3.2 Représentation de l'erreur de positionnement

On peut noter $\rho(\vec{u}_i, \vec{u}_j)$ comme étant la corrélation entre les vecteurs $\vec{u}_i$ et $\vec{u}_j$, ce qui est en fait le produit scalaire des vecteurs unitaires. Ceci permet de définir l'erreur de positionnement relative entre les deux vecteurs $\vec{u}_i$ et $\vec{u}_j$ associés aux intervalles mélodiques de i et j demi-tons comme étant :

$$\varepsilon_{ij} = \rho(\vec{u}_i, \vec{u}_j) - H(i-j) \tag{3.10}$$

L'erreur est donc la différence entre la corrélation harmonique désirée $H(i-j)$ et la corrélation présente des deux vecteurs d'intervalle mélodique.

Le but recherché est de minimiser l'erreur totale de l'ensemble des vecteurs d'intervalle mélodique. L'erreur totale peut s'écrire sous la forme :

$$\begin{aligned} E &= \frac{1}{2}\sum_{i=-(M-1)/2}^{(M-1)/2}\sum_{j=-(M-1)/2}^{(M-1)/2}\varepsilon_{ij}^2 \\ &= \frac{1}{2}\left|U^TU - H\right|^2 \end{aligned} \tag{3.11}$$

La matrice H est la matrice de corrélation désirée et ses éléments sont déterminés par la fonction de corrélation harmonique : $H_{ij} = H(i-j)$. Il s'agit donc d'une matrice symétrique.

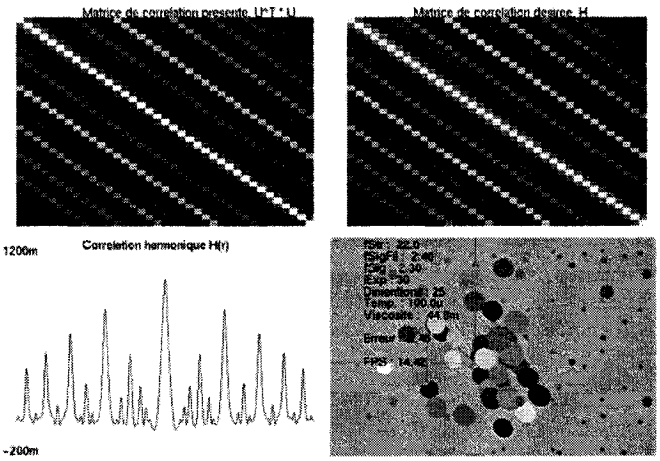

FIG. 3.5 Positions des vecteurs d'intervalle mélodique à l'équilibre dans un espace de 25 dimensions. En haut, à gauche, est donnée la matrice de corrélation présente U^TU entre les positions des vecteurs. En haut, à droite, est donnée la matrice de corrélation désirée H. En bas, à gauche, est illustrée la corrélation harmonique $H(r)$. En bas, à droite, sont illustrées les positions des vecteurs dans l'espace.

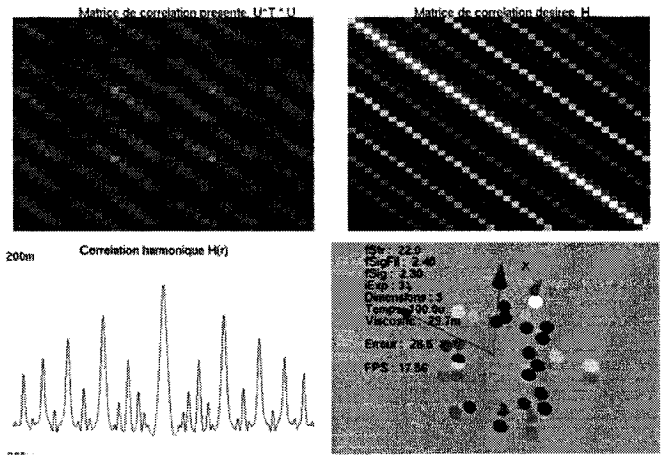

FIG. 3.6 Positions des vecteurs d'intervalle mélodique à l'équilibre dans un espace de 3 dimensions.

3.3.3 Descente du gradient

Afin de minimiser l'erreur, on effectuera la descente du gradient de manière itérative jusqu'à ce que l'erreur devienne négligeable.

$$\begin{aligned} \Delta U &= -\beta \nabla E \\ &= -\beta U \left(U^T U - H \right) \end{aligned} \tag{3.12}$$

Ici, β est le taux d'apprentissage. Il doit être ni trop petit, sinon on convergerait lentement vers un minimum, ni trop grand, sinon on perdrait la convergence. Une bonne heuristique à utiliser (Bishop, 1995) est de mettre à jour β en l'incrémentant si l'erreur a diminué après la dernière itération ou en le diminuant si l'erreur a augmenté après la dernière itération.

$$\beta_{nouveau} = \begin{cases} \rho \beta_{vieux} & si\ \Delta E < 0, \\ \sigma \beta_{vieux} & si\ \Delta E > 0. \end{cases} \tag{3.13}$$

Le paramètre ρ est un peu plus grand que l'unité ($\rho = 1.05$) et le paramètre σ est plus petit que l'unité ($\sigma = 0.5$).

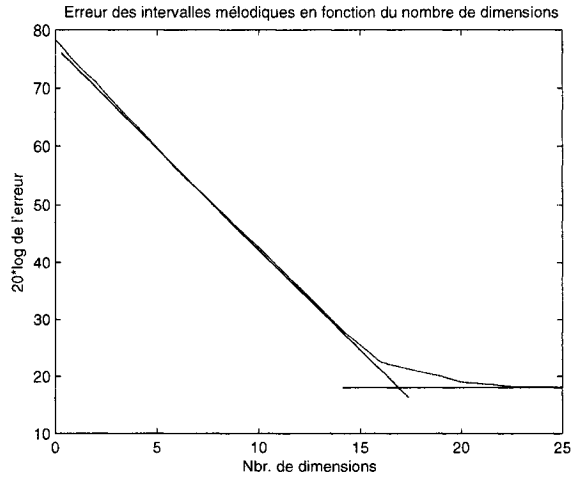

FIG. 3.7 Erreur à l'équilibre en fonction du nombre de dimensions pour 37 intervalles mélodiques.

3.4 Résultats de recherche d'intervalles mélodiques vectoriels

Pour effectuer la descente du gradient[5], nous avons choisi un ensemble de $M = 37$ intervalles mélodiques espacés d'un demi-ton chacun. Cela couvre la plage des intervalles allant de la onzième[6] descendante à la onzième ascendante, soit un peu plus que de l'octave descendant à l'octave ascendant. Ceci permet de couvrir amplement les intervalles mélodiques présents dans les chansons usuelles. Les vecteurs $\vec{u}_r$ des intervalles mélodiques sont initialisés dans l'espace $\mathbb{R}^N$ selon une distribution normale $N(0, \frac{1}{\sqrt{N}})$. En laissant les points se positionner, on obtient une matrice de corrélation $U^T U$ entre les positions des vecteurs associés aux intervalles mélodiques respectant la corrélation désirée (voir figure 3.5).

Selon la dimensionnalité de l'espace $\mathbb{R}^N$ dans lequel les vecteurs d'intervalle mélodique évoluent, il est possible d'atteindre une corrélation plus ou moins proche de la corrélation désirée. À la figure 3.5, on a utilisé $N = 25$ dimensions, tandis qu'à

[5]Ceci a été réalisé en programmation $C++$.

[6]Une onzième est un intervalle de 18 demi-tons.

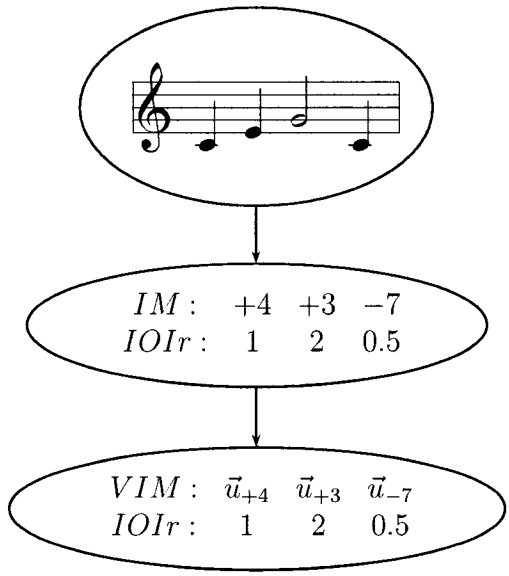

FIG. 3.8 Extraction des vecteurs d'intervalle mélodique, VIM, et des ratios d'intervalle temporel, IOIr, à partir de la séquence de notes obtenue.

la figure 3.6, on a utilisé $N = 3$ dimensions seulement. Comme on peut l'observer, la corrélation obtenue avec 3 dimensions est loin d'être celle désirée. Il faut donc chercher le nombre optimal de dimensions pour avoir la corrélation désirée.

Vient un nombre de dimensions où il devient superflu d'en ajouter. Il s'agit en fait du seuil où les vecteurs d'intervalle mélodique ont suffisamment de dimensions pour se positionner de manière à respecter la corrélation harmonique. Il est possible d'observer, à la figure 3.7, la variation de l'erreur en fonction du nombre de dimensions.

On observe qu'à partir de 17 dimensions, l'erreur semble atteindre un plancher. Dans notre cas, nous avons utilisé 25 dimensions pour exprimer les vecteurs $\vec{u}_r$ associés aux intervalles mélodiques de notre séquence mélodique.

3.5 Conclusion du chapitre

La représentation vectorielle des intervalles mélodiques permet de respecter la structure mélodique que présente une séquence de notes musicales. La séquence, initialement représentée par une série de paires de nombres réels (un pour l'intervalle mélodique et un pour le ratio de durées de notes) voit ses intervalles mélodiques remplacés par des vecteurs unitaires en N dimensions $\vec{u}_i$ (voir figure 3.8). Ceci permet de respecter les proximités harmoniques.

Dans le chapitre suivant, nous verrons comment il est possible d'évaluer le degré de ressemblance entre deux séquences de notes décrites par leurs VIM $\vec{u}_i$ et leurs IOIr.

CHAPITRE 4

ANALYSE DE SÉQUENCES MUSICALES AU MOYEN D'UN RÉSEAU À ÉCHO

Lorsqu'un usager donne à l'ordinateur un air musical qu'il possède en mémoire, l'information ne sera pas nécessairement fidèle à l'air de référence. Ainsi, l'usager pourrait, par exemple, se souvenir seulement d'un extrait quelconque situé à un temps arbitraire à l'intérieur de l'air de référence. Il pourrait chanter une série de thèmes de la chanson dans un ordre arbitraire. Il pourrait aussi chanter des variations mélodiques n'apparaissant pas dans la chanson originale (Dannenberg et al., 2003). On peut toutefois espérer que certains motifs musicaux soient similaires à l'air recherché.

En effet, si les conditions dans lesquelles sont données les séquences musicales n'empêchent pas un auditeur humain de reconnaître une chanson, elles deviennent plus problématiques lorsqu'on veut reconnaître les séquences à l'aide d'un ordinateur. Il faut donc trouver un moyen de reconnaître les motifs contenus dans les chansons fournies.

Dans ce chapitre, nous aborderons la méthode utilisée permettant de reconnaître les mélodies fournies par un usager. Cette dernière utilise un réseau de neurones de type réseau à écho ou ESN[1]. Un réseau de neurones est une fonction mathématique discriminante. On retrouve également l'appellation machine à état liquide ou LSM[2] qui constitue une variante du réseau à écho. Pour débuter, nous décrirons brièvement deux architectures de réseaux de neurones que sont les perceptrons et

[1]Echo state network (ESN).
[2]Liquid state machine (LSM).

les réseaux récurrents de Hopfield. Ensuite, nous décrirons les réseaux à écho qui constituent une classe particulière de réseaux récurrents. Finalement, nous décrirons leur utilisation pour la reconnaissance de séquences musicales.

4.1 Réseau de neurones

On définit un réseau de neurones (Haykin, 1999) comme un ensemble d'éléments de calcul non linéaires densément interconnectés. Ces éléments de calcul sont dénommés neurones pour l'analogie avec les cellules du système nerveux. L'opération effectuée par un neurone est décrite à la figure 4.1.

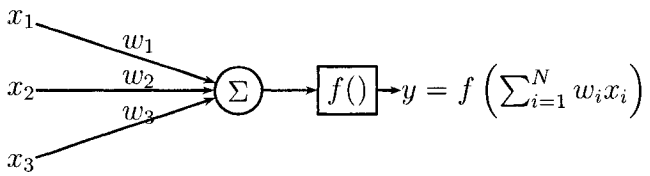

FIG. 4.1 Neurone formel.

On emploie généralement pour fonction de sortie $f()$ une sigmoïde telle que $\tanh(ax)$, $\frac{1}{1+e^{-\alpha x}}$ ou encore une simple fonction seuil :

$$f(x) = \begin{cases} +1 & \text{si } x \geq 0, \\ -1 & \text{si } x < 0. \end{cases}$$

4.2 Perceptron multicouche

Dans un réseau à propagation vers l'avant ou perceptron (à une ou plusieurs couches), chaque couche j de neurones, notée $\vec{x}_j$, est fonction de la couche précédente $\vec{x}_{j-1}$ (voir figure 4.2). Cela peut s'exprimer à l'aide de la matrice des poids

d'interconnexion $\mathbf{W}^{j-1,j}$ entre les neurones des couches $j-1$ et j de la façon suivante :

$$\vec{x}_j = f\left(\mathbf{W}^{j-1,j}\vec{x}_{j-1}\right) \tag{4.1}$$

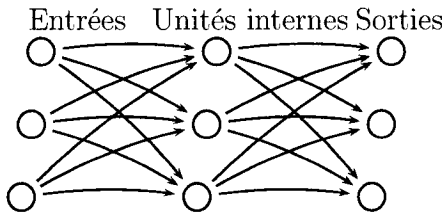

FIG. 4.2 Perceptron multicouche.

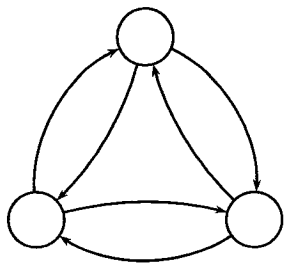

FIG. 4.3 Réseau de neurones récurrent de Hopfield à trois neurones.

4.3 Réseau de neurones récurrent de Hopfield

Un réseau de neurones récurrent de Hopfield (Haykin, 1999) est un ensemble de neurones à rétroaction où l'état futur de chaque neurone dépend de l'état présent des autres neurones (voir figure 4.3).

Ainsi, dans un réseau de neurones récurrent, tel que celui de la figure 4.3, l'évolution de l'état interne $\vec{x}(n+1)$ sera fonction de l'état précédent $\vec{x}(n)$. Ceci peut s'exprimer

avec la matrice $\mathbf{W}$ d'interconnexions entre ces neurones internes par l'expression suivante :

$$\vec{x}(n+1) = f\left(\mathbf{W}\vec{x}(n)\right) \tag{4.2}$$

Une propriété importante des réseaux de Hopfield (Rabiner et Juang, 1993; Haykin, 1999) est que si la matrice des poids $\mathbf{W}$ est symétrique ($w_{ij} = w_{ji}$), l'état interne du réseau tendra alors vers un état stable où $\vec{x}(n) = \vec{x}(n-1)$. Un réseau de Hopfield peut ainsi contenir un ensemble d'états stables appelés attracteurs. Il s'agit des points où l'énergie du système est minimisée et vers où tend l'état du système.

Ces points de relaxation représentent des configurations stables du réseau et peuvent être utilisés sous la forme de mémoire associative. On peut, en effet, fixer les points de relaxation sur M points d'intérêt $\vec{x}_\mu$ (pouvant représenter un ensemble de caractères à reconnaître) en ajustant les poids w_{ij}. L'ajustement des poids s'écrit avec un ensemble de N neurones, M points d'intérêt et $\mathbf{I}$ la matrice identité comme étant :

$$\mathbf{W} = \frac{1}{N}\sum_{\mu=1}^{M} \vec{x}_\mu \vec{x}_\mu^T - M\mathbf{I} \tag{4.3}$$

En initialisant l'état $\vec{x}(n_0)$ proche d'un état $\vec{x}_\mu$ et en appliquant l'équation 4.2, on aura $\vec{x}(n)$ qui tendra vers l'état de repos $\vec{x}_\mu$. Cela permet d'associer à un vecteur d'entrée $\vec{x}(n_0)$ le vecteur $\vec{x}_\mu$.

4.4 Réseau à écho

Dans le cas du réseau à écho (réseau de neurones récurrent), il est possible de travailler avec un seul état interne stable et attracteur, soit l'état de repos $\vec{x} = \mathbf{0}$. Il s'agit de l'état vers lequel tend le réseau à écho lorsque l'on cesse de le stimuler. Dans ce cas, on ne cherche plus à utiliser les états attracteurs comme mémoire du

système. On considère plutôt le réseau de neurones comme un système dynamique dans lequel il devient possible d'enregistrer l'historique des entrées passées dans les perturbations qui lui sont imposées. L'analogie que l'on peut faire est celle d'un liquide où tombent des gouttes et où des ondelettes témoignent des perturbations apportées au bassin d'eau, d'où l'appellation de la variante machine à état liquide (LSM) du réseau à écho.

Dans ce type de système dynamique, on peut considérer que les perturbations apportées au système construisent une trajectoire d'états transitoires internes. Cette trajectoire caractérisera la séquence des perturbations apportées au système. C'est dans ce sens que l'on utilisera le réseau de neurones récurrent.

De plus, ce type de réseau récurrent est de haute dimension (une centaine ou plus de neurones interconnectés), ce qui permet une meilleure séparabilité des motifs. Comme le dit le théorème de Cover sur la séparabilité des motifs, qui peut, en termes qualitatifs, être cité comme suit (Cover, 1965) :

- un problème de classification de motifs transposé dans un espace de haute dimension de façon non linéaire est plus propice à être séparable linéairement que dans un espace de basse dimension.

Dans le cas de la reconnaissance de mélodies, les perturbations apportées au système seront les couples $\langle VIM, IOIr \rangle$. Ainsi, chaque note de la séquence stimulera le réseau à écho. Ceci permettra alors d'enregistrer la séquence musicale dans les états internes du réseau à écho.

Notre hypothèse est que si l'extrait est suffisamment long, il devrait pouvoir exciter le réseau à écho dans une série d'états proches d'états rencontrés auparavant dans une mélodie semblable. Ce rapprochement entre l'état actuel du système et un état déjà rencontré sera détecté par une mémoire associative.

De plus, puisque les intervalles mélodiques de l'extrait musical sont transformés en vecteurs respectant la corrélation harmonique (VIM), on suppose que le système gardera une stimulation semblable, tant que la mélodie principale sera respectée.

4.5 Description du réseau à écho utilisé

Pour réaliser ce système dynamique, on emploiera un réseau de neurones récurrent de haute dimension, donc un grand nombre de neurones internes interconnectés. Le réseau de neurones à écho utilisé (Jaeger, 2001), dans le cadre de ce projet, se définit comme étant un réseau récurrent possédant au temps t :

1. un vecteur d'entrée $\vec{u}(n)$ de dimension K, donc K entrées $u_i(n)$;
2. un vecteur d'état interne $\vec{x}(n)$ de dimension N, donc N neurones $x_j(n)$ internes ;
3. un vecteur de sortie $\vec{y}(n)$ de dimension L, donc L sorties $y_k(n)$.

Les poids des interconnections du système sont décrits par différentes matrices de connexion :

1. le vecteur d'entrée $\vec{u}(n)$ excite les unités internes $x_i(n)$ selon la matrice de connexion d'entrée $\mathbf{W}^{in}_{N\times K}$;
2. les unités internes $\vec{x}(n)$ sont interconnectées suivant la matrice $\mathbf{W}_{N\times N}$;
3. les unités de sortie $\vec{y}(n)$ prennent leur valeur des unités internes $x_i(n)$ selon la matrice de connexion de sortie $\mathbf{W}^{out}_{L\times N}$.

Il s'agit d'un cas particulier des réseaux à écho décrit par H. Jaeger. Ici, les sorties ne sont connectées qu'aux neurones internes.

Ces interconnections sont illustrées à la figure 4.4. À noter que seuls les poids $\mathbf{W}^{out}$ des unités de sortie sont ajustables, les deux autres matrices de poids sont fixées.

Il est donc possible d'ajuster la matrice des poids de sortie $\mathbf{W}^{out}$ afin d'avoir la lecture désirée.

Un bon réseau à écho doit présenter de bons comportements dynamiques entre les neurones internes. On peut d'ailleurs qualifier un réseau à écho de réservoir riche en comportements dynamiques (Jaeger, 2001). Pour obtenir ce type de comportements, on choisit une matrice d'interconnexion interne $\mathbf{W}$ creuse. Ce faible taux de connexion entre les neurones internes permet de découper le réseau en sous-réseaux, ce qui encourage les dynamiques locales. En pratique, le remplissage de 3 à 5 % de la matrice permet d'obtenir le comportement désiré d'une ESN.

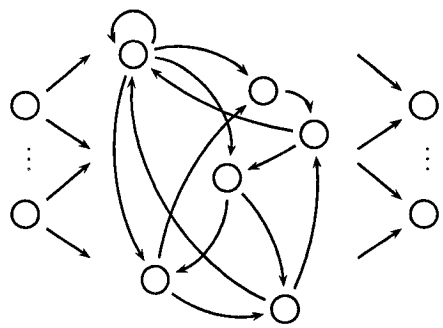

FIG. 4.4 Réseau à écho (ESN).

On décrit l'évolution de l'état des neurones internes du réseau à écho utilisé comme étant fonction de l'état présent des neurones internes et des entrées.

$$\vec{x}(n+1) = f\left(\mathbf{W}^{in}\vec{u}(n+1) + \mathbf{W}\vec{x}(n)\right) \tag{4.4}$$

La fonction $f()$ est une fonction sigmoïde. Dans notre cas, on a utilisé la fonction tanh().

La sortie du système, pour sa part, sera une lecture pondérée de l'état actuel du système :

$$\vec{y}(n+1) = f\left(\mathbf{W}^{out}\vec{x}(n)\right) \tag{4.5}$$

4.5.1 Notation compacte des séquences

Pour avoir une notation plus compacte, il sera pratique de décrire les matrices donnant les séquences des différents vecteurs d'état du système : $\vec{u}(n), \vec{x}(n), \vec{y}(n)$ ainsi que $\vec{d}(n)$ qui est la sortie désirée au temps n. Celles-ci donneront l'historique de l'évolution du système. Pour une séquence de longueur h, on notera ces matrices :

$$\begin{aligned}
\bar{\mathbf{u}}^h &= (\vec{u}(0), \vec{u}(1), \ldots, \vec{u}(h-1)) \\
\bar{\mathbf{x}}^h &= (\vec{x}(0), \vec{x}(1), \ldots, \vec{x}(h-1)) \\
\bar{\mathbf{y}}^h &= (\vec{y}(0), \vec{y}(1), \ldots, \vec{y}(h-1)) \\
\bar{\mathbf{d}}^h &= \left(\vec{d}(0), \vec{d}(1), \ldots, \vec{d}(h-1)\right)
\end{aligned}$$

Une autre notation utilisée pour rendre le langage plus compact est l'emploi de l'opérateur T. Ce dernier permet de décrire l'impact d'une séquence $\bar{\mathbf{u}}^h$ à l'entrée sur l'état $\vec{x}$ du système. On notera : $\vec{x}(n+h) = T(\vec{x}(n), y(n), \bar{\mathbf{u}}^h)$ pour dénoter l'évolution de l'état interne du système après h temps et la séquence d'entrée $\bar{\mathbf{u}}^h$.

4.5.2 Génération des états internes du ESN

En partant d'une séquence mélodique[3] décrite par une séquence de couples $\langle VIM, IOIr\rangle_n$ (voir section 3.5), on génère les différents états internes $\vec{x}(n)$ du réseau à écho comme suit :

[3] Aussi bien pour les chansons de la collection que pour les chansons fournies par les locuteurs.

1. Fournir à l'entrée du ESN le premier couple $\langle VIM, IOIr \rangle_0$ de la séquence musicale.
2. Générer le nouvel état de la ESN suivant selon l'équation 4.4.
3. Passer au prochain intervalle mélodique de la chanson.

4.6 Propriétés d'un réseau à écho

Un ESN possède les propriétés suivantes le rendant propice à la reconnaissance de séquences temporelles :

1. Propriété de séparation
2. Propriété d'approximation
3. Mémoire évanescente

Ces propriétés permettent, dans le cadre de la reconnaissance de séquences musicales, d'extraire les motifs musicaux associés à une chanson.

4.6.1 Propriété de séparation

La propriété la plus importante d'un ESN émergeant de l'analyse théorique et des simulations sur ordinateur est sa capacité à réagir différemment à différentes séquences fournies en entrée en créant différentes trajectoires d'états internes. On nomme cette propriété la propriété de séparation (SP[4]). Elle permet de quantifier la séparation entre deux trajectoires d'états internes du système, causée par deux séries d'entrées différentes. En reprenant l'analogie du liquide, il s'agit de la dif-

[4]Separation Property (SP).

férence des motifs des vagues résultant des différentes perturbations apportées au liquide.

4.6.2 Propriété d'approximation

De la propriété de séparation découle la propriété d'approximation. Celle-ci est la capacité que possèdent les neurones de sortie de pouvoir reconnaître différentes séquences $\bar{\mathbf{x}}$ d'états internes pour y associer la sortie désirée. Ceci implique donc que l'on puisse faire apprendre à un réseau à écho à reconnaître des motifs temporels, comme on va le voir à la section 4.8.

4.6.3 Mémoire évanescente

Une des propriétés fondamentales d'un ESN qui permettent d'identifier des schémas temporels est sa mémoire évanescente. Cette dernière est la capacité du réseau à oublier les états passés et elle permet, dans le cas de la reconnaissance de séquences musicales, d'identifier des schémas musicaux à court terme.

Étant donné les différents états initiaux de l'état interne du système $\vec{x}_1(0)$ et $\vec{x}_2(0)$ et la séquence $\bar{\mathbf{u}}^h$ de h perturbations, on observe que $d\left(T(\vec{x}_1(0), \bar{\mathbf{u}}^h), T(\vec{x}_2(0), \bar{\mathbf{u}}^h)\right)$ sera d'autant plus petite que h sera grand. $T(\vec{x}_1(0), \bar{\mathbf{u}}^h)$ représente l'état interne $\vec{x}_1(h-1)$ du réseau à écho après avoir été soumis à la série $\bar{\mathbf{u}}^h$ de h perturbations.

Ce phénomène, appelé contraction d'état, est illustré à la figure 4.5. Il y est modélisée l'évolution de différentes séquences d'états internes du réseau à écho. Ces séquences sont notées $\bar{\mathbf{x}}_1$, $\bar{\mathbf{x}}_2$ et $\bar{\mathbf{x}}_3$. Chacune est initialisée à une position quelconque. Cependant, étant donné qu'elles possèdent toutes la même séquence de perturbations $\bar{\mathbf{u}}^h$, les différentes séquences tendront vers une trajectoire commune.

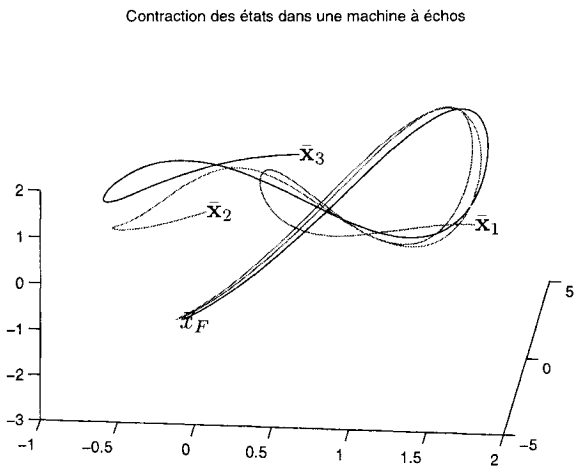

FIG. 4.5 Représentation du phénomène de contraction des états dans un réseau à écho.

De cette manière, l'état dans lequel se trouvait le réseau à écho est oublié et n'influence plus l'état présent du réseau. C'est ce que l'on appelle la mémoire évanescente d'un réseau à écho. Cette mémoire peut être représentée par la demi-vie du réseau. La demi-vie $T_{1/2}$ est le nombre moyen d'itérations h après lequel la distance $d\left(T(\vec{x}_1(0), \bar{\mathbf{u}}^h), T(\vec{x}_2(0), \bar{\mathbf{u}}^h)\right)$ a diminué de moitié par rapport à la distance initiale $d\left(\vec{x}_1(0), \vec{x}_2(0)\right)$.

4.7 Stabilité et dynamisme d'un réseau à écho

Pour avoir ces propriétés de séparation et d'approximation des ESN, il est important qu'un réseau possède une bonne dynamique lui permettant de générer différents chemins d'états internes. Toutefois, un réseau à écho doit avoir des poids d'interconnexion bien ajustés pour ne pas tomber en régime chaotique.

4.7.1 Stabilité d'un réseau à écho

Pour qu'un réseau à écho soit stable, il faut qu'il tende vers l'état de repos, soit $\vec{x} = \mathbf{0}$, lorsqu'il n'est pas stimulé. Ainsi, contrairement à un réseau de Hopfield où les points attracteurs sont associés à la mémoire du réseau, dans un réseau à écho, il n'y a que le point de repos qui soit attracteur. La mémoire du réseau est inscrite implicitement dans l'état actuel $\vec{x}(n)$ du réseau qui est fonction des perturbations passées. Ces perturbations (entrées $\vec{u}(n)$) créent une trajectoire $\bar{\mathbf{x}}$ caractéristique de la séquence de perturbations.

Pour observer la propriété de mémoire évanescente, on doit avoir un seul point attracteur, sinon des suites de perturbations semblables pourraient créer différentes trajectoires $\bar{\mathbf{x}}$ d'états internes du réseau à écho (par exemple, en étant pris à proximité de différents attracteurs). Pour respecter cette contrainte, on doit observer la fonction d'énergie (Lyapunov) qui s'écrit comme celle d'un réseau de Hopfield (Haykin, 1999) :

$$E = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij}x_i x_j + \sum_{j=1}^{N}\int_0^{x_j} \tanh^{-1}(x)dx \tag{4.6}$$

w_{ij} sont les poids de la matrice $\mathbf{W}$.

Pour que l'état $\vec{x} = \mathbf{0}$ soit le point de repos, il doit être le minimum d'énergie unique du système. Ceci peut se produire tant que les poids d'interconnexion ne sont pas trop élevés, c'est-à-dire tant que le rayon spectral de la matrice d'interconnexion est plus petit que 1 (Jaeger, 2001). On définit le rayon spectral comme suit :

$$\rho(\mathbf{W}) = \max(|\lambda_i|) \tag{4.7}$$

λ_i sont les valeurs propres de la matrice $\mathbf{W}$. Dans le cas où $\rho(\mathbf{W}) > 1$, l'état $\vec{x} = \mathbf{0}$

devient un point de répulsion.

Par contre, si le rayon spectral $\rho(\mathbf{W})$ est trop faible, c'est-à-dire s'il y a peu de stimulations entre les neurones internes, le réseau tend rapidement vers son point de repos. On a alors une demi-vie très faible et par le fait même une courte mémoire. Dans ce cas, le réseau à écho ne présente pas de dynamiques intéressantes et ses performances diminues (voir section 6.3.1).

4.8 Reconnaissance de motifs à l'aide d'un réseau à écho

Comme on l'a vu, il est possible de connecter des neurones de sortie au réseau à écho pour reconnaître certains motifs temporels fournis à l'entrée. On pourrait ainsi imaginer des neurones de sortie permettant de détecter les différents thèmes musicaux des chansons de notre collection (voir annexe I). On pourrait également entraîner des neurones de sortie pour obtenir la prochaine note. Cela permettrait, en branchant la sortie à l'entrée, de répéter indéfiniment la séquence musicale.

Dans cette section, nous verrons comment il est possible d'entraîner des neurones de sortie en modifiant la matrice des poids de sortie $\mathbf{W}^{out}$ afin de détecter des schémas temporels fournis au ESN. Nous verrons également une seconde méthode plus directe, consistant à utiliser une mémoire associative pour reconnaître les états internes du réseau à écho pour y associer les chansons de la collection.

4.8.1 Entraînement des poids de sortie

Afin de trouver la matrice des poids idéaux permettant d'avoir la sortie désirée, on utilisera la méthode de Gauss-Newton. Cette dernière permet d'optimiser $\mathbf{W}^{out}$.

On suppose donc que les observations ont été recueillies dans l'intervalle $n = \{0, \ldots, h-1\}$. L'erreur peut être considérée comme étant fonction des poids de sortie. Avec $\bar{\mathbf{d}}^h$ (l'historique des valeurs désirées pour h observations successives), l'historique des erreurs $\bar{\mathbf{e}}^h$ à la sortie s'écrit :

$$\bar{\mathbf{e}}^h(\mathbf{w}^{out}) = \tanh^{-1}(\bar{\mathbf{d}}^h) - \mathbf{w}^{out}\bar{\mathbf{x}}^h \tag{4.8}$$

On recherche le minimum d'énergie par une approximation de Taylor :

$$\bar{\mathbf{e}}^h(\mathbf{w}^{out}) = \bar{\mathbf{e}}^h(\mathbf{w}_0^{out}) + \left(\mathbf{w}^{out} - \mathbf{w}_0^{out}\right) \nabla_{\mathbf{w}^{out}} \bar{\mathbf{e}}^h(\mathbf{w}^{out})\Big|^T_{\mathbf{w}^{out}=\mathbf{w}_0^{out}} \tag{4.9}$$

Avec le Jacobien :

$$J = \nabla \bar{\mathbf{e}}^h(\mathbf{w}^{out})^T \tag{4.10}$$

et quelques simplifications de notation, on peut réécrire 4.9 sous la forme :

$$\bar{\mathbf{e}} = \bar{\mathbf{e}}_0 + \Delta \mathbf{w} J \tag{4.11}$$

On cherche ensuite à minimiser le carré des erreurs :

$$\frac{1}{2}|\bar{\mathbf{e}}|^2 = \frac{1}{2}|\bar{\mathbf{e}}_0|^2 + J\bar{\mathbf{e}}_0^T \Delta \mathbf{w} + \frac{1}{2} J J^T \Delta \mathbf{w}^T \Delta \mathbf{w} \tag{4.12}$$

en mettant la différentielle à zéro :

$$\mathbf{0} = \bar{\mathbf{e}}_0 J^T + \Delta \mathbf{w} J J^T \tag{4.13}$$

Avec l'erreur définie comme étant :

$$\bar{\mathbf{e}}_0 = \tanh^{-1}(\bar{\mathbf{d}}) - \mathbf{w}_0 \bar{\mathbf{x}} \tag{4.14}$$

on a alors :

$$J = \nabla \bar{\mathbf{e}}_0 = -\bar{\mathbf{x}} \tag{4.15}$$

On trouve une bonne évaluation pour la matrice de poids de sortie :

$$\begin{aligned} \mathbf{0} &= \bar{\mathbf{e}}_0 \bar{\mathbf{x}}^T - \Delta \mathbf{w} \bar{\mathbf{x}} \bar{\mathbf{x}}^T \\ \mathbf{0} &= \left(\tanh^{-1}(\bar{\mathbf{d}}) - \mathbf{w}_0 \bar{\mathbf{x}}\right) \bar{\mathbf{x}}^T - \left(\mathbf{w} - \mathbf{w}_0\right) \bar{\mathbf{x}} \bar{\mathbf{x}}^T \\ \mathbf{w} &= \tanh^{-1}(\bar{\mathbf{d}}) \bar{\mathbf{x}}^T \left(\bar{\mathbf{x}} \bar{\mathbf{x}}^T\right)^{-1} \\ \mathbf{w} &= \tanh^{-1}(\bar{\mathbf{d}}) \bar{\mathbf{x}}^\dagger \end{aligned} \tag{4.16}$$

Lorsque l'on cherche à faire l'entraînement des poids de sortie, il est important d'ajouter du bruit à l'entrée. Ceci permet d'avoir des poids de sortie qui donnent un signal stable à la sortie (Jaeger, 2001).

4.8.2 Association des états internes générés

Au lieu de brancher des neurones de sortie et d'évaluer les poids de sortie $\mathbf{W^{out}}$, on peut utiliser une méthode plus directe pour identifier les schémas d'une séquence musicale. Puisque l'état interne du ESN décrit déjà de façon implicite la séquence des dernières perturbations, on pourrait utiliser une mémoire associative (voir chapitre 5) pour associer aux états internes du réseau à écho les chansons de la collection.

Pour ce faire, on suppose que l'on possède une séquence temporelle de h stimuli $\vec{u}(n)$, $n = 0, ..., h-1$ noté $\bar{\mathbf{u}}^h$. Dans notre cas, les stimulis sont les couples $\langle VIM, IOIr \rangle$ (voir section 3.5). Cette séquence $\bar{\mathbf{u}}^h$ génère une trajectoire $\bar{\mathbf{x}}^h$ de h états internes $\vec{x}(n)$ la caractérisant. On peut ainsi associer directement ces états $\vec{x}(n)$ générés à la séquence de stimuli $\bar{\mathbf{u}}^h$. On enregistre alors dans la mémoire as-

sociative le nom de la séquence $\bar{\mathbf{u}}^h$ (le titre de la chanson) pour chacun des états internes qu'elle a générés.

4.9 Conclusion du chapitre

Dans ce chapitre, nous avons vu comment un ESN peut être utilisé pour reconnaître des séquences temporelles grâce à sa mémoire évanescente. En observant simplement les états internes générés par le réseau et en les associant à la mélodie les ayant générés, il est possible de trouver cette mélodie en générant des états semblables. Pour ce faire, nous utiliserons une mémoire associative permettant de dire pour chaque état $\vec{x}(n)$ généré par un locuteur quelles sont les chansons qui y sont associées. Dans le prochain chapitre, nous verrons comment construire ce type de mémoire.

CHAPITRE 5

RECHERCHE DE SÉQUENCES ASSOCIÉES

Dans le chapitre précédent, nous avons vu comment il est possible de caractériser une séquence musicale à l'aide d'un réseau à écho. Nous nous attarderons maintenant au problème de la reconnaissance d'une séquence musicale ayant généré une séquence $\bar{\mathbf{x}}$ d'états internes du ESN. Cette reconnaissance se fait à l'aide de la mémoire hétéro-associative. Pendant l'entraînement, cette mémoire associe à chaque état interne $\vec{x}(n)$ pouvant être généré par le réseau à écho des titres de chansons (chaînes de caractères) de la collection de chansons (voir figure 5.1).

Pour la reconnaissance, on compare les états générés $\vec{x}(n)$ par la séquence $\bar{\mathbf{u}}$ avec les états enregistrés $\vec{x}_i$ lors de l'entraînement. Ceci permet, en cumulant les titres de chansons associés aux états $\vec{x}_i$ les plus proches des états $\vec{x}(n)$ générés, de déterminer la chanson ayant le plus de chance d'avoir été chantée (voir figure 3 dans l'introduction).

Le problème est donc de rechercher les états $\vec{x}_i$ enregistrés se rapprochant le plus des états générés $\vec{x}(n)$ par le réseau à écho. Le nombre d'états enregistrés étant d'autant plus grand que le nombre de chansons de la collection est grand, il est important de classer ces états enregistrés (environ 1 200 pour les 36 chansons). Ceci évite de devoir chercher parmi tous les états enregistrés dans la mémoire. Pour effectuer cette classification, on aura recours à des arbres de recherche.

Dans ce chapitre, nous commencerons par décrire la mémoire associative utilisée. Nous décrirons par la suite les propriétés particulières des espaces de haute dimension, car les états internes $\vec{x}(n)$ du réseau à écho que l'on cherche à reconnaître

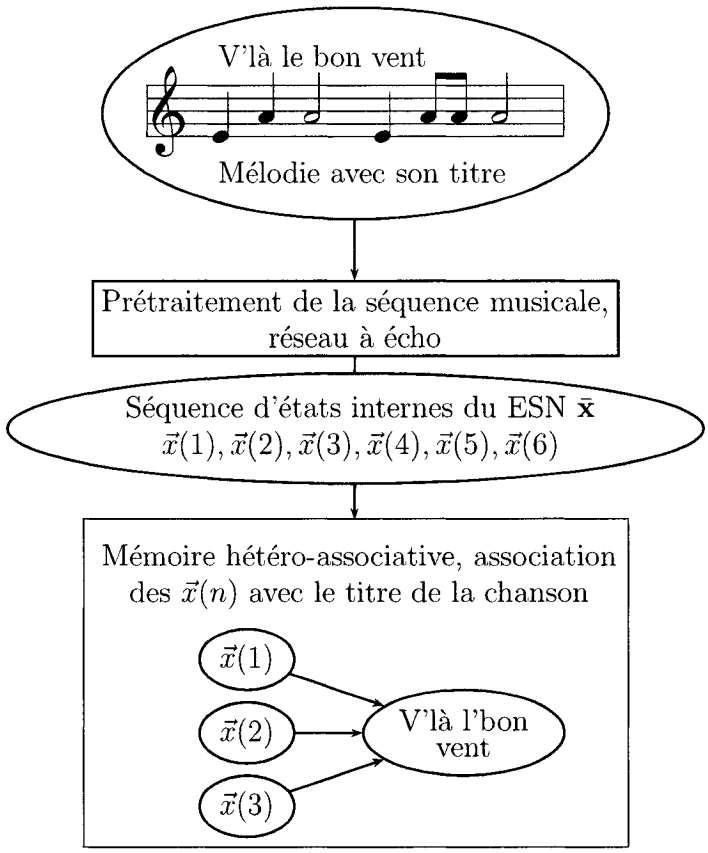

FIG. 5.1 Entraînement de la mémoire hétéro-associative.

sont de haute dimension (plus de 100 neurones internes). Finalement, on décrira les méthodes efficaces de recherche dans les espaces de haute dimension grâce aux arbres de recherche.

5.1 Description de la mémoire associative

Une mémoire associative[1] se définit comme une structure reliant un ensemble de motifs d'entrée à un ensemble de motifs de sortie. Il existe deux types de mémoire associative : la mémoire auto-associative et la mémoire hétéro-associative. Dans une

[1]http ://www.comp.nus.edu.sg/~pris/AssociativeMemory/AssociativeMemoryContent.html

mémoire auto-associative tel le réseau de Hopfield (voir section 4.3), la mémoire trouve un motif ayant été enregistré et se rapprochant le plus du motif fourni à l'entrée. Dans une mémoire hétéro-associative, le motif récupéré à la sortie est, de façon générale, différent de celui présenté à l'entrée, non seulement par son contenu mais également par sa nature.

Une mémoire associative peut être comparée avec la mémoire humaine parce qu'elle a la capacité de reconnaître de larges motifs (ou idées) et qu'elle est très tolérante à l'erreur. En effet, les signaux reçus dans une région du cerveau à divers moments peuvent varier grandement, sans pour autant être mal interprétés. Ainsi, la mémoire cérébrale peut effectuer des associations presque instantanément afin de transmettre le bon message. C'est cette capacité à faire des associations qui fait sa force et qui permet une abstraction des signaux fluctuants.

La mémoire hétéro-associative utilisée pour reconnaître les états du ESN en les associant à des chansons de la collection (voir annexe I) est constituée de deux ensembles. Le premier ensemble **A** (**A**dresses de la mémoire) contient les points d'intérêt. Ceux-ci sont des états générés par le ESN et sont notés $\vec{x}_\mu \in [-1, 1]^N$. Le deuxième ensemble **C** (**C**ontenu de la mémoire) contient les données (titres des chansons) associées aux points d'intérêt (états $\vec{x}_\mu$ du ESN).

5.2 Propriétés des espaces de haute dimension

Les espaces de haute dimension possèdent la propriété particulière d'avoir un grand nombre de voisins possibles. Par exemple, si l'on considère des empilements compacts de sphères, en 1D, un point posséderait 2 voisins, en 2D, 6 voisins, en 3D, 12 voisins, en 4D, 20 voisins et en N-D, il posséderait $N(N+1)$ voisins. De plus, le

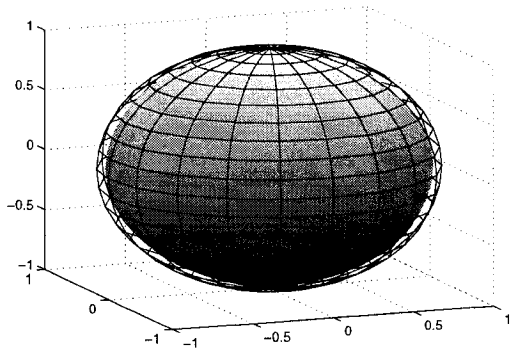

FIG. 5.2 Représentation d'une sphère et de sa croute avec $\varepsilon = 0.05$.

volume d'une sphère de rayon r dans un espace en N dimensions augmente selon :

$$V_{\text{sphere}} = \begin{cases} \frac{1}{(N/2)!}\pi^{\frac{N}{2}} r^N & \text{si } N \text{ est pair,} \\ \frac{2^N((N-1)/2)!}{N!}\pi^{\frac{N-1}{2}} r^N & \text{si } N \text{ est impair,} \end{cases} \tag{5.1}$$

que l'on peut simplifier par :

$$V_{\text{sphere}} = \alpha(N) r^N \tag{5.2}$$

avec :

$$\alpha(N) = \begin{cases} \frac{1}{(N/2)!}\pi^{\frac{N}{2}} & \text{si } N \text{ est pair,} \\ \frac{2^N((N-1)/2)!}{N!}\pi^{\frac{N-1}{2}} & \text{si } N \text{ est impair,} \end{cases} \tag{5.3}$$

On peut observer que si l'on distribue des points uniformément dans cette sphère de rayon r et de dimension N, avec un N grand, la plus grande partie des points se situeront dans la croute de l'hypersphère. Ceci étant dû au fait que le volume varie en r^N. Pour le démontrer, on considère le ratio de volume entre le volume d'une croute d'épaisseur εr par rapport au volume total de la sphère :

$$\frac{\alpha(N) r^N - \alpha(N)\left(r(1-\varepsilon)\right)^N}{\alpha(N) r^N} = 1 - (1-\varepsilon)^N \tag{5.4}$$

Ceci donnerait, par exemple (voir figure 5.2), pour un $N = 100$ et un $\varepsilon = \frac{1}{20}$ (soit une croute d'épaisseur du vingtième du rayon de la sphère) une croute contenant $99,4$ % du volume total de la sphère.

De plus, une caractéristique de ces vecteurs de haute dimension est que lorsque l'on choisit deux points au hasard sur la surface d'une hypersphère, ils seront généralement non corrélés[2] entre eux. Ainsi, si l'on prend les deux pôles d'une hypersphère de rayon R situés sur l'axe z et que l'on observe la distribution des points positionnés uniformément à l'intérieur de cette hypersphère projetée sur l'axe z, on aurait pour l'équation de la sphère :

$$\begin{aligned} z^2 + r^2 &= R^2 \\ r &= \sqrt{R^2 - z^2} \end{aligned} \tag{5.5}$$

Cela donne une distribution du volume selon l'axe z de :

$$\begin{aligned} dV &= \alpha(N-1)r^{N-1}dz \\ dV &= \alpha(N-1)\left(R^2 - z^2\right)^{\frac{N-1}{2}} dz \end{aligned} \tag{5.6}$$

Pour avoir une idée, cette fonction possède l'allure donnée à la figure 5.3 pour $R = 1$ et $N = 100$.

5.2.1 Variance de la distribution du volume d'une hypersphère

Une valeur importante est la variance de la distribution du volume de l'hypersphère. Cette dernière sera utile lors de la construction des arbres de recherche. La variance

[2]La corrélation $\rho(p_1, p_2) \in [-1, 1]$ entre deux points prend la valeur 1 lorsque les deux vecteurs sont les mêmes, -1 lorsqu'ils sont opposés et 0 quand ils sont perpendiculaires ou non corrélés. Elle se définit ici comme le produit scalaire normalisé.

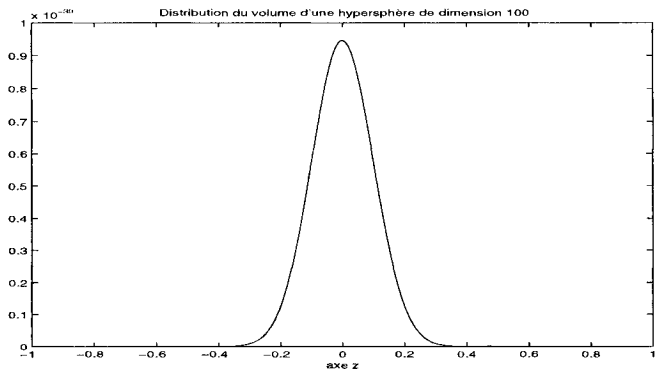

FIG. 5.3 Distribution du volume suivant un axe reliant deux pôles d'une hypersphère de rayon 1 et de dimension 100.

se définit comme :

$$\begin{aligned} \sigma^2 &= \frac{\int_{-R}^{R} z^2 dV}{\int_{-R}^{R} dV} \\ \sigma^2 &= \frac{\int_{-R}^{R} z^2 \alpha(N-1)\left(R^2-z^2\right)^{\frac{N-1}{2}} dz}{\int_{-R}^{R} \alpha(N-1)(R^2-z^2)^{\frac{N-1}{2}} dz} \\ \sigma^2 &= \frac{R^2}{N+2} \end{aligned} \tag{5.7}$$

On a donc un écart type de :

$$\sigma = \frac{R}{\sqrt{N+2}} \tag{5.8}$$

On peut voir que lorsque N devient grand, si l'on choisit deux points au hasard sur la surface de l'hypersphère, il y de fortes chances que ceux-ci soient non corrélés, c'est-à-dire que $\vec{x}_1 \cdot \vec{x}_2 \simeq 0$. Chacun des points de l'hypersphère sera donc non corrélé avec la plupart des autres points.

Cependant, si l'on prend quelques points non corrélés (des points quelconques à la surface de l'hypersphère), par exemple $\vec{x}_1$, $\vec{x}_2$ et $\vec{x}_3$, il est possible de trouver un point intermédiaire $\vec{x}_{int}$ à ces points qui sera proche de tous les points en même temps. Ce point sera tout simplement la moyenne normalisée des points et contiendra l'information sur chacun de ses constituants. Par exemple, en prenant trois points, le point intermédiaire aurait une corrélation avec les points de $\rho(\vec{x}_{int}, \vec{x}_i) = \frac{1}{\sqrt{3}} \simeq 0,577$. En observant la distribution du volume d'une hyper-

sphère, à la figure 5.3, on remarque que très peu de points peuvent se trouver entre $0,577$ et 1. Cette caractéristique signifie que l'espace environnant le point intermédiaire borné par ces autres points (l'hypersphère contenant les autres points) est très petit par rapport à l'espace total.

5.3 Arbre de recherche

Pour trouver efficacement une chanson, nous devons identifier pour chacun des états internes $\vec{x}(n)$ générés par le réseau à écho les chansons pouvant être associées à ces états. Pour faire ces recherches, nous employons les techniques de recherche par arborescence. Ces techniques permettent de trouver le plus rapidement possible les plus proches voisins d'un point.

Ainsi, un problème éventuel avec les mémoires associatives est le temps de recherche des locations les plus proches d'un vecteur d'entrée $\vec{x}$. En effet, il s'avère très coûteux de vérifier tous les éléments de l'ensemble $\mathbf{A}$ des points d'intérêt $\vec{x}_\mu$.

Ainsi, l'utilisation des arbres de recherche (Liu et al., 2004) pourrait permettre l'accélération de l'accès aux points d'intérêt $\vec{x}_\mu$ ayant la plus grande affinité avec un vecteur d'entrée $\vec{x}(n)$. De façon générale, l'arbre de recherche accélère la recherche en évoluant dans un arbre binaire où il compare à chaque branche le vecteur d'entrée avec les différents pivots contenus dans chaque nœud de l'arbre. Ceci permet, en partant du nœud racine, de réduire son champ de recherche à chaque branche visitée pour aboutir à un nœud contenant une liste de feuilles où se trouvent des points d'intérêt $\vec{x}_\mu$ ayant le plus de chance d'être proche du vecteur d'entrée $\vec{x}(n)$.

Nous aborderons dans cette section deux types d'arbres particulièrement bien adaptés pour la classification des données. Il s'agit des arbres métriques et des arbres étendus.

L'idée principale des arbres de recherche tels les arbres métriques ou les arbres étendus est d'effectuer une série de séparations de l'espace avec des plans de séparation (dichotomies) jusqu'à ce que la région finale soit suffisamment limitée.

5.3.1 Arbres métriques

Les arbres métriques (Liu et al., 2004) sont simples à construire et permettent la réalisation d'une structure efficace pour la recherche des k plus proches voisins. Les points d'un ensemble y sont organisés de manière hiérarchique.

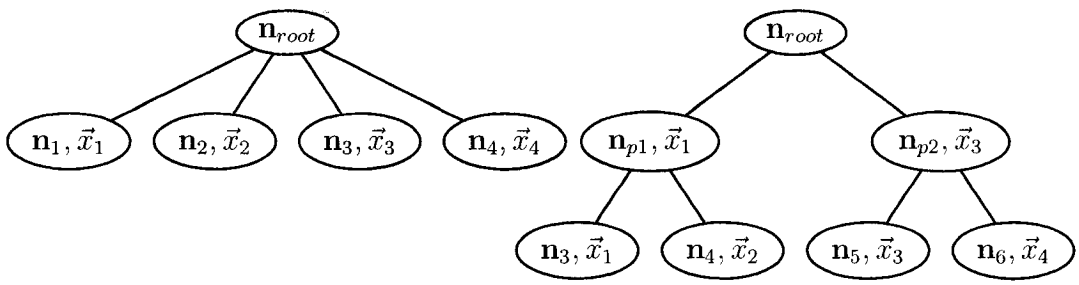

FIG. 5.4 Construction d'un arbre métrique.

On débute en prenant la racine de l'arbre $\mathbf{n}_{root}$ qui représente l'ensemble total $\mathbf{A}$ des N points $\vec{x}_\mu$. L'ensemble des points est présent à la racine comme un ensemble $\mathbf{D}$ de descendants. Cet ensemble sera divisé en deux sous-ensembles contenant chacun une partie des points de l'ensemble de départ. L'arbre métrique est donc un arbre binaire. La séparation des sous-ensembles s'effectue récursivement jusqu'à ce que les nœuds descendants générés contiennent des sous-ensembles ne contenant pas plus que S_{max} éléments.

5.3.1.1 Classement des points

Pour effectuer la séparation du nœud $\mathbf{n}$ contenant un ensemble $\mathbf{D}$ de descendants en deux sous-ensembles, on détermine les pivots $\mathbf{n}_{p1}$ et $\mathbf{n}_{p2}$ de ce nœud. Ces pivots deviendront les nouveaux descendants directs du nœud $\mathbf{n}$ et s'en partageront les feuilles (voir figure 5.4). Les pivots d'un nœud se définissent comme étant les deux feuilles du nœud $\mathbf{n}$ les plus éloignées, ce qui se traduit par :

$$[\mathbf{n}_{p1}, \mathbf{n}_{p2}] = \underset{\mathbf{n}_j, \mathbf{n}_k \in \mathbf{D}}{\operatorname{argmax}} \left(d(\mathbf{n}_j, \mathbf{n}_k) \right). \tag{5.9}$$

$d(\mathbf{n}_j, \mathbf{n}_k) = |\vec{x}_j - \vec{x}_k|$ est la distance euclidienne entre les deux points.

Pour trouver les deux feuilles les plus éloignées, on commence par trouver le descendant le plus éloigné du premier descendant. Ce dernier sert de premier pivot. Ensuite, on cherche le point le plus éloigné du premier pivot qui servira de second pivot.

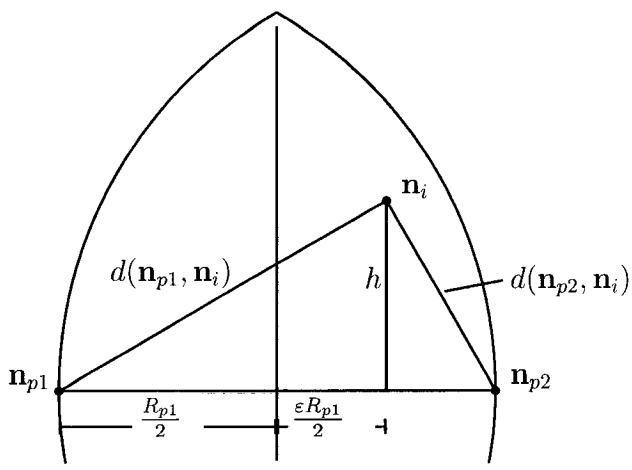

FIG. 5.5 Classement d'un point dans la lentille de dichotomie.

Pour classer les points, on utilise la variable d'indice de classement $\varepsilon \in [-1, 1]$,

présentée à la figure 5.5. C'est l'indice de la distance du point à classer $\mathbf{n}_i$ par rapport au plan de séparation créé par les deux pivots. La variable ε est négative lorsque le point $\mathbf{n}_i$ est plus proche du premier pivot et elle est positive lorsque le point est plus proche du second pivot. Ainsi, on aura :

$$\varepsilon \begin{cases} \leq 0 & \mathbf{n}_i \text{ appartient } \mathbf{n}_{p1}, \\ > 0 & \mathbf{n}_i \text{ appartient } \mathbf{n}_{p2}. \end{cases} \tag{5.10}$$

Pour trouver ε, il suffit de remarquer, à la figure 5.5, que :

$$\begin{aligned} h^2 + \left(\tfrac{(1+\varepsilon)R_{p1}}{2}\right)^2 &= d(\mathbf{n}_{p1}, \mathbf{n}_i)^2 \\ h^2 + \left(\tfrac{(1-\varepsilon)R_{p1}}{2}\right)^2 &= d(\mathbf{n}_{p2}, \mathbf{n}_i)^2 \end{aligned} \tag{5.11}$$

D'où l'on peut tirer :

$$\varepsilon = \frac{d(\mathbf{n}_{p1}, \mathbf{n}_i)^2 - d(\mathbf{n}_{p2}, \mathbf{n}_i)^2}{R_{p1}^2} \tag{5.12}$$

Il est à noter que la dichotomie en deux pivots crée une lentille connue sous le nom de vesica piscis[3], dans laquelle tous les sous-éléments du nœud initial sont inclus. Cette lentille, nous la nommons lentille de dichotomie puisque c'est à l'intérieur de celle-ci que s'effectue la dichotomie.

5.3.1.2 Recherche dans l'arbre métrique

Pour effectuer la recherche des points se rapprochant d'un point de référence $\mathbf{n}_{ref}$, on effectue une recherche en profondeur d'abord ou DFS[4]. On commence par la racine de l'arbre métrique et l'on utilise l'indice de classement ε pour déterminer le pivot à visiter en premier. On accumule ainsi, en explorant l'arbre, une liste des

[3]La lentille vesica piscis est l'intersection de deux hypersphères (dans notre cas, mais usuellement deux cercles) de même rayon et dont le centre de chacune des hypersphères se situe à la surface de l'autre.

[4]Deep first search (DFS).

plus proches voisins retenus que l'on met dans la liste des descendants du point de référence.

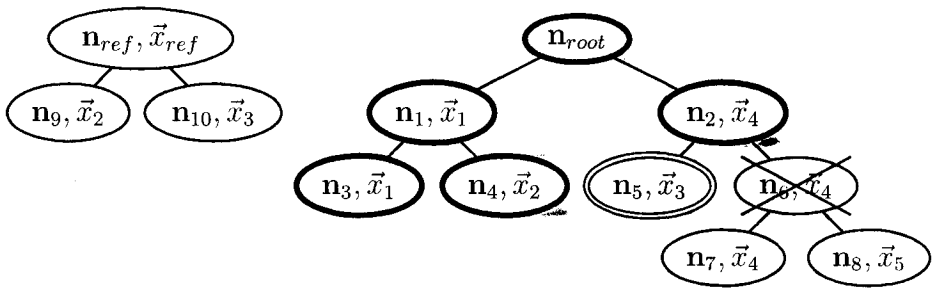

FIG. 5.6 Recherche en profondeur d'abord.

On peut éviter de rechercher dans un nœud lorsque l'on s'aperçoit qu'aucun des points inclus dans ce nœud ne peut être plus proche que les plus proches voisins retenus.

5.3.2 Arbres étendus

Pour la recherche dans les espaces de haute dimension, il y a un type d'arbre qui est plus intéressant que les arbres métriques : ce sont les arbres étendus ou spill-tree (Liu et al., 2004). En effet, bien que les arbres métriques soient excellents lorsque le nombre de dimensions n'est pas trop élevé, soit moins que vingt, ils deviennent plutôt médiocres lorsque le nombre de dimensions augmente. Il n'est alors pas rare de devoir chercher toute l'arborescence pour trouver le point le plus proche du point de référence.

L'arbre étendu fonctionne en général de la même manière que l'arbre métrique. Sauf que, dans le cas d'un arbre étendu, on tolère un certain dédoublement des sous-ensembles des pivots. Lors de la construction des pivots, au lieu de classer les points d'un côté ou de l'autre du plan de dichotomie selon l'équation 5.13, on tolère

une certaine marge $\tau \in [-1, 1]$ quant à l'indice de classement.

$$\varepsilon \begin{cases} \leq \tau & \mathbf{n}_i \text{ appartient } \mathbf{n}_{p1}, \\ > -\tau & \mathbf{n}_i \text{ appartient } \mathbf{n}_{p2}, \end{cases} \tag{5.13}$$

Les points ayant un $-\tau \leq \varepsilon < \tau$ se trouvent donc dans les deux pivots.

Ceci permettra, dans le cas des arbres de recherche binaires, d'effectuer une recherche défaitiste[5] en ayant tout de même de bons résultats.

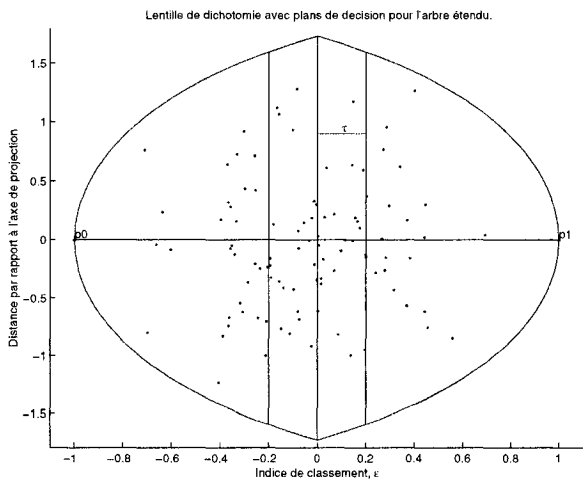

FIG. 5.7 Séparation des points entre deux pivots pour un arbre étendu.

Dans le cas où l'on doit effectuer des recherches sur les plus proches voisins du point de référence $\mathbf{n}_{ref}$ dans un grand nombre de dimensions, il est plus avantageux de déborder de la médiane, étant donné que la plupart des points se trouvent aux environs de la médiane (voir figure 5.7). On évite alors d'éliminer des points proches du point de lecture $\mathbf{n}_{ref}$ se trouvant dans le voisinage de la médiane.

Une façon de déterminer le seuil τ de décision lors de la classification des points dans l'arbre étendu est de se baser sur l'écart type de la distribution des points projetés

[5]Une recherche défaitiste est une recherche en profondeur d'abord dans laquelle on ne vérifie pas si les autres branches permettent de trouver un point plus proche.

sur l'axe des deux pivots (indice de classement ε). En haute dimension, on s'attend normalement à ce que la plupart des points se trouvent autour de la médiane avec un certain écart type. Cet écart type s'exprime comme étant $\sigma = \frac{1}{\sqrt{N+2}}$ (voir section 5.2). On peut alors poser :

$$\tau \propto \frac{1}{\sqrt{N+2}} \tag{5.14}$$

5.3.2.1 Arbres hybrides

Cependant, les arbres étendus peuvent parfois présenter une très grande profondeur, et ce, dû au fait que l'on peut garder une grande partie des points à chaque séparation au lieu de diviser les sous-ensembles de moitié comme dans les arbres métriques. Pour remédier à ce problème, on utilise les arbres hybrides. Ces derniers utilisent des nœuds étendus sauf dans le cas où les points inclus dans la région $-\tau \leq \varepsilon < \tau$ dépassent un certain ratio nommé « seuil de balance » de l'ordre de $0,5$. Dans le cas où plus de la moitié des points seraient communs aux deux pivots, il serait préférable de refaire les pivots avec une séparation métrique ($\tau = 0$) pour conserver une profondeur raisonnable. On marquerait alors ce nœud comme étant métrique et l'on y ferait une recherche en profondeur d'abord non défaitiste.

5.3.2.2 Projections en basse dimension

Bien que les arbres étendus soient plus efficaces que les arbres métriques pour effectuer des recherches dans un espace de haute dimension, ils demeurent tout de même affectés par la « malédiction des hautes dimensions » (Liu et al., 2004). En effet, lorsque la dimension est élevée, on a besoin d'un τ suffisamment élevé pour ne pas éliminer des points potentiellement intéressants, mais on perd alors l'efficacité

de la recherche puisque l'on garde trop de points à chaque séparation de l'ensemble de recherche. On a alors recours au théorème de Johnson-Lindenstrauss qui affirme que :

Pour $\forall \epsilon \in [0,1]$ avec $k \geq 4\left(\frac{\epsilon^2}{2} - \frac{\epsilon^3}{3}\right)^{-1} \ln(n)$ et pour un ensemble $\mathbf{W}$ de n points dans $\mathbb{R}^d$, il existe une transposition $f : \mathbb{R}^d \mapsto \mathbb{R}^k$ telle que pour $\forall u$ et $v \in \mathbf{W}$ on a :

$$(1-\epsilon)\,|u-v|^2 \leq |f(u)-f(v)|^2 < (1+\epsilon)\,|u-v|^2 \tag{5.15}$$

Cela signifie qu'il est possible de transposer un ensemble de points d'un espace de haute dimension d vers un espace de plus basse dimension k sans qu'il y ait une distorsion importante dans la distance entre les points.

Une manière d'effectuer la transposition est de multiplier simplement l'ensemble des N points de départ $\mathbf{A}_{d\times N}$ par une matrice de transposition $\mathbf{T}_{k\times d}$ ayant pour éléments des valeurs distribuées de manière aléatoire selon une distribution de moyenne nulle et de variance de 1. On peut naturellement penser à une loi normale $N(0,1)$. Un autre choix pour la matrice de transposition est la matrice d'Achlioptas (Achlioptas, 2001). Il s'agit de mettre des 1 et des -1 avec une probabilité de $\frac{1}{2}$ chacun dans $\mathbf{T}_{k\times d}$.

Une fois la transposition des points vers un espace de basse dimension effectuée, on peut y construire un arbre hybride. Cependant, la distorsion de la transposition entraîne des erreurs dans le recherche des plus proche voisins. Pour minimiser cette erreur, on crée L matrices de transposition $\mathbf{T}$ permettant d'avoir L ensembles de basse dimension (avec L arbres hybrides de recherche) associés à l'ensemble de départ $\mathbf{A}$. Ainsi, si la probabilité d'échec de recherche d'un plus proche voisin dans un des arbres hybrides de basse dimension est δ, cette probabilité diminue à δ^L avec les projections multiples.

5.4 Cumul des chansons retenues

Afin de déterminer la chanson ayant le plus de chance d'être associée à la mélodie chantée, on effectue la cumulation des titres retenus. On commence par initialiser le ESN en le stimulant avec les 3 premiers vecteurs d'entrée $\vec{u}(0)$, $\vec{u}(1)$ et $\vec{u}(2)$. On ignore ainsi les premiers états générés par le ESN. Ceci lui évite d'essayer de reconnaître des motifs de 2 ou 3 intervalles mélodiques arbitraires pouvant appartenir à n'importe quelle mélodie.

On enregistre ensuite, pour chacun des $h-3$ états $\vec{x}(n)$ ($n \in \{3, 4, ..., h-1\}$) générés par la séquence $\bar{\mathbf{u}}^h$, les k titres les plus prometteurs, c'est-à-dire les k titres associés aux k plus proches voisins de chaque état $\vec{x}(n)$. À ces k titres retenus pour chacun des états $\vec{x}(n)$, on attribue un poids allant en décroissant (un poids de k pour le plus proche, $k-1$ pour le 2^e plus proche, etc.)(voir figure 5.8).

En cumulant les poids sur les h états, on crée la liste des chansons ayant le plus de chance d'être associées à la chanson chantée. À cette liste sont associés les poids de chacun des titres des chansons.

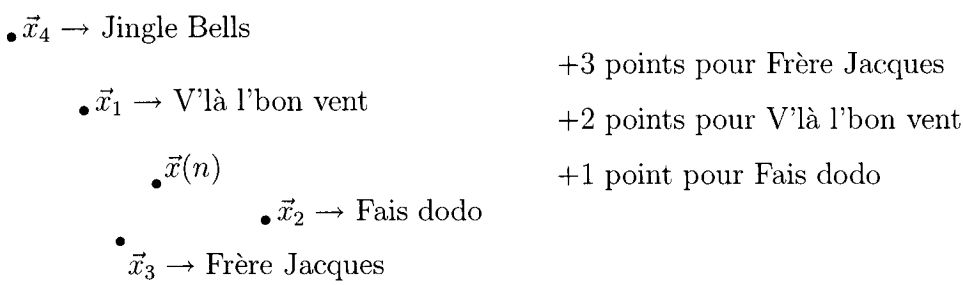

FIG. 5.8 Méthode du cumul : pour chaque état $\vec{x}(n)$ généré par le ESN, on ajoute des points pour les k (ici $k = 3$) chansons associées aux états $\vec{x}_\mu$ ayant la plus grande affinité avec les états $\vec{x}(n)$.

Puisque le ESN possède une mémoire évanescente, les états $\vec{x}(n)$ ne dépendent que des dernières notes qui viennent de stimuler le réseau. En utilisant cette technique,

on a donc de bonnes chances de reconnaître une séquence, et ce, même lorsqu'elle débute à un instant arbitraire ou que les thèmes de la séquence sont désordonnés.

5.5 Conclusion du chapitre

Dans ce chapitre, nous avons vu l'utilité de la mémoire hétéro-associative pour trouver les chansons ressemblant à la chanson ayant été chantée. Le système de reconnaissance de séquences musicales est maintenant décrit dans son ensemble et on peut maintenant regarder son fonctionnement et ses performances.

CHAPITRE 6

PROGRAMME DE RECONNAISSANCE DE SÉQUENCES MUSICALES ET RÉSULTATS

Dans les chapitres précédents, on a abordé les différents sujets permettant de réaliser un système de reconnaissance de séquences musicales. Dans ce chapitre, on décrit le programme réalisé permettant la reconnaissance de séquences musicales chantées. On décrit également les performances que donne ce programme dans l'identification des séquences. La mesure employée pour évaluer ces performances est l'inverse du taux de classement moyen ou MRR[1] (Dannenberg et al., 2003; Adams, 2004).

Si τ_k est la position de classement pour la requête k, on définit le MRR pour N requêtes comme :

$$MRR = \frac{1}{N} \sum_{k=1}^{N} \frac{1}{\tau_k} \tag{6.1}$$

Les extraits de chansons ont été recueillis chez quelques 15 chanteurs ayant accepté de chanter un total de 130 extraits de chansons comprises dans la collection de 36 chansons (voir annexe I). On a obtenu pour ces extraits un MRR global de 81 % et 74 % des extraits ont été retrouvés en première position.

[1]Mean Reciprocal Rank (MRR).

6.1 Description du programme de reconnaissance de séquences musicales

Dans le cadre de ce mémoire, un programme de reconnaissance de séquences musicales a été réalisé dans le langage de programmation C++, dans un environnement OpenGL[2]. Ce programme comporte une interface usager et permet d'effectuer l'acquisition du signal et son prétraitement en temps réel.

Une personne désirant effectuer la recherche d'une mélodie n'a qu'à chanter un court extrait de la mélodie à l'aide d'un microphone. Une fois l'acquisition du signal sonore terminée, la personne peut alors visualiser la séquence de notes déterminée par le programme. De cette séquence de notes chantée, on peut effectuer une recherche dans la collection de chansons et trouver les 5 chansons qui se rapprochent le plus de la séquence chantée (voir figures 3.4, 6.10 et 6.11).

En plus d'effectuer l'acquisition du signal, le programme permet de faire jouer la mélodie chantée pour s'assurer que la machine a bien compris. Le programme permet également d'éditer la séquence de notes retenue. On peut effacer des notes, en insérer, les aligner sur la même tonalité et en modifier la durée ou la fréquence. Ceci permet naturellement à un usager d'ajuster sa mélodie pour qu'elle soit plus facile à trouver.

Finalement, le programme permet de faire jouer les mélodies de la collection de chansons.

En plus de ce programme, des bancs d'essais ont également été réalisés pour obtenir les résultats des tests réalisés dans les sections qui suivent.

[2]Le programme contient plus de 5000 lignes de code.

6.2 MRR en fonction des paramètres de segmentation

Les premiers tests effectués consistent à déterminer les meilleurs paramètres pour effectuer la segmentation des notes. Ces paramètres, vus aux sections 2.3.1 et 2.3.2, sont : Δt_{min} le paramètre d'anti-rebond des notes et Δf_{plt} la hauteur des plateaux associés aux notes.

6.2.1 Anti-rebond

On présente, à la figure 6.1, le MRR en fonction du temps minimal d'une note Δt_{min} (voir section 2.3.2). Pour trouver le MRR et son écart type à la figure 6.1a, ainsi qu'aux figures suivantes, on évalue d'abord le MRR pour chacun des chanteurs (figure 6.1b), puis on évalue ensuite la moyenne et l'écart type des différents chanteurs .

On observe que la durée minimale Δt_{min} optimale est de 3 fenêtres de découpage, soit $138ms$. Bien que $138ms$ soit un temps optimal pour la plupart des chanteurs, 3 des chanteurs obtiennent un meilleur MRR lorsque Δt_{min} est de l'ordre de $275ms$. Il pourrait s'agir de chanteurs hésitant sur les attaques de notes (le début d'une attaque serait alors interprété comme une petite note de fréquence différente de la note réelle). On pourrait alors penser modifier le programme pour qu'il s'adapte à ce type de chanteurs en proposant les résultats pour un Δt_{min} plus long.

6.2.2 Hauteur des plateaux

On présente, à la figure 6.2, le MRR en fonction de la hauteur des plateaux Δf_{plt} (voir section 2.3.1).

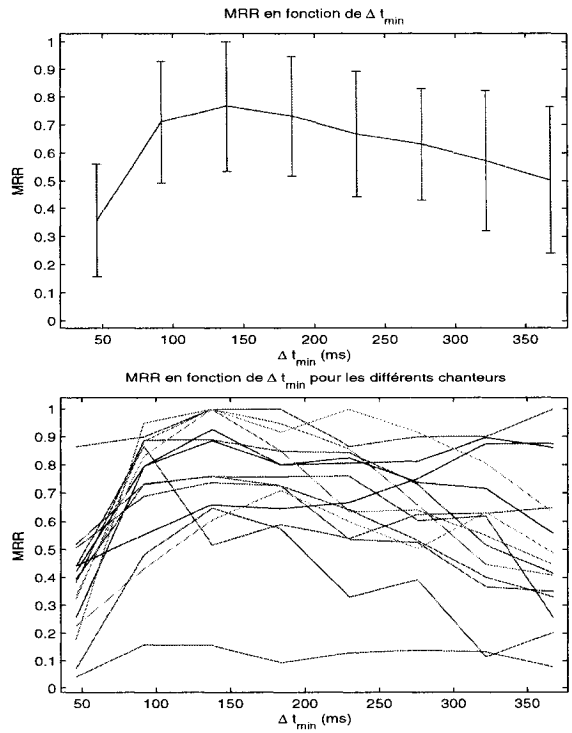

FIG. 6.1 Anti-rebond a) MRR en fonction de Δt_{min}. b) MRR en fonction de Δt_{min} pour les 15 chanteurs.

On peut observer que la hauteur des plateaux Δf_{plt} n'est pas critique. Toutefois, une hauteur d'environ 0.85 demi-ton permet d'avoir un MRR optimal.

6.3 Réseau à écho

Après la segmentation, l'élément le plus critique du système est le réseau à écho qui est l'élément clé de la reconnaissance des motifs musicaux. On a effectué une série de tests pour évaluer les poids et le taux de connexion optimum de la matrice d'interconnexion du réseau à écho $\mathbf{W}$.

Le réseau à écho utilisé contient 128 neurones internes, et la matrice de connexion d'entrée $\mathbf{W}^{in}$ demeure constante avec des poids $w_{ij}^{in} = \pm 0.3$ (probabilité de $\frac{1}{2}$ de

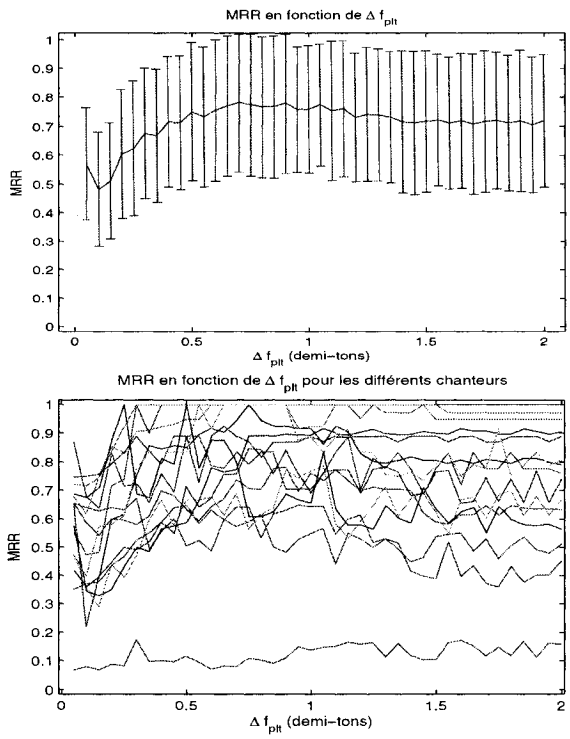

FIG. 6.2 Hauteur des plateaux de note a) MRR en fonction de Δf_{plt}. b) MRR en fonction de Δf_{plt} pour les 15 chanteurs.

valoir, soit 0.3 ou -0.3). Les vecteurs d'entrée $\vec{u}(n)$ du réseau à écho sont les couples $\langle VIM, IOIr \rangle$.

La matrice d'interconnexion du réseau à écho $\mathbf{W}_{128\times128}$ est creuse avec un taux de connexion de 5 %. Les poids non nuls sont fixés à une valeur de $w_{ij} = \pm w$ (probabilité de $\frac{1}{2}$ de valoir, soit w ou $-w$).

6.3.1 Force des poids d'interconnexion

Le premier test fait sur le réseau à écho porte sur l'impact du rayon spectral $\rho(\mathbf{W})$ de la matrice $\mathbf{W}$ d'interconnexion du réseau (voir figure 6.3). La constance des

poids w est choisie afin d'avoir le rayon spectral $\rho(\mathbf{W})$ désiré. On note que $\rho(\mathbf{W})$ est directement proportionnel à w (voir section 4.7.1).

Dans le cas où $\mathbf{W}$ a un taux de remplissage de 5 %, on a un $\rho(\mathbf{W}) = 1$ lorsque $w = 0,38 \pm 0,01$.

a)

b)

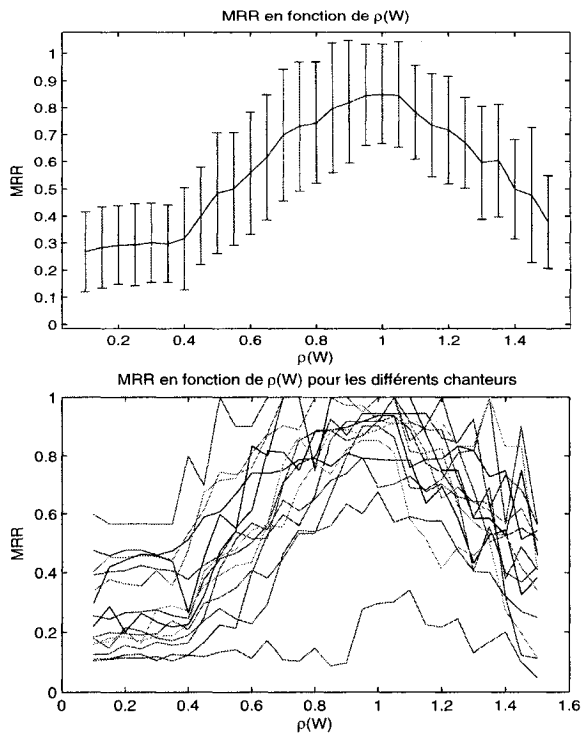

FIG. 6.3 Rayon spectral $\rho(\mathbf{W})$ de la matrice de poids du réseau à écho a) MRR en fonction de $\rho(\mathbf{W})$. b) MRR en fonction de $\rho(\mathbf{W})$ pour les différents chanteurs.

On observe, comme l'a fait H. Jeager, que le système est optimal quand le rayon spectral est juste au-dessous de 1. Au-delà de 1, le réseau à écho perd de la stabilité et devient moins performant. Dans notre système, un rayon spectral de 0.9 a été choisi. Ceci donne des poids de $w = 0,38 \times 0,90 = 0,34$.

On a également observé que la demi-vie $T_{1/2}$ (voir section 4.6.3) du réseau variait avec la force des poids d'interconnexion (voir tableau 6.1). Plus la force des poids

$\rho(\mathbf{W})$	$T_{1/2}$
0.50	0.9
0.75	1.7
0.90	2.9
1.00	4.7

TAB. 6.1 Tableau de la demi-vie $T_{1/2}$ en nombre d'itérations du réseau à écho pour différents rayons spectraux $\rho(\mathbf{W})$.

est élevée, plus le réseau a une forte mémoire. Il y a alors plus de dynamique interne dans le réseau à écho.

6.3.2 Taux de remplissage de la matrice d'interconnexion

Le second test réalisé sur le réseau à écho permet d'observer l'impact du taux de remplissage de la matrice d'interconnexion. On a ajusté les poids afin de conserver un rayon spectral d'environ $\rho(\mathbf{W}) \simeq 0,90$.

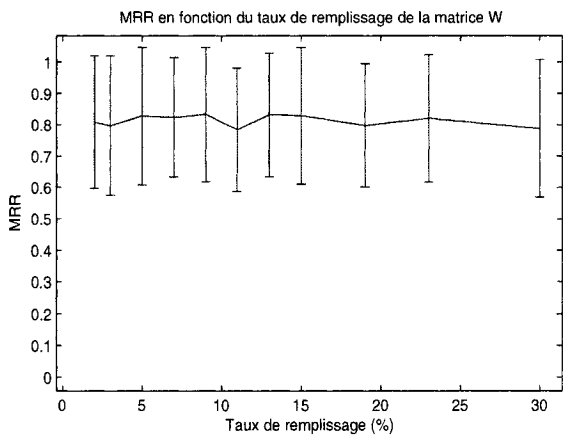

FIG. 6.4 MRR en fonction du taux de remplissage de la matrice d'interconnexion $\mathbf{W}$ du réseau à écho.

Bien que le taux de remplissage de la matrice $\mathbf{W}$ n'influence pas grandement les performances du système (voir figure 6.4), nous avons arrêté notre choix sur un

réseau ayant un taux de connexion de 5 %. On peut noter que d'avoir un faible taux de connexion évite de faire trop de calculs pour l'état futur du réseau à écho.

6.4 Propriétés de la mémoire héréro-associative

6.4.1 Rayon minimal de la mémoire hétéro-associative

Un facteur important influençant les performances du programme de reconnaissance de séquences musicales est la distance minimale R_{min} acceptée entre deux états internes du réseau à écho $\vec{x}_1$ et $\vec{x}_2$ enregistrés dans la mémoire hétéro-associative. Étant donné que les thèmes musicaux ont tendance à se répéter à l'intérieur d'une même chanson, le réseau à écho recevra des séries de perturbations $\bar{\mathbf{u}}$ associées aux thèmes. Ceci a pour effet, à cause du phénomène de contraction d'état (voir section 4.6.3), de créer des états internes semblables pour les thèmes qui se répètent à l'intérieur d'une même chanson (voir figure 6.4.1).

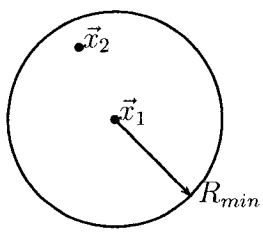

FIG. 6.5 Deux états proches du réseau à écho générés par la même série de notes.

Si deux chansons ont un thème commun et stimulent, par le fait même, des états semblables $\vec{x}_i$ dans le réseau à écho, on associe alors les deux titres de chansons à ces états $\vec{x}_i$. À la figure 6.6, on observe l'impact de R_{min} sur le MRR. Plus R_{min} est grand, plus la mémoire est petite. Ceci évite de prendre trop d'espace mémoire lorsque la collection de chansons devient importante. De plus, avoir une distance minimale trop petite oblige à enregistrer plusieurs fois le même thème d'une chan-

a)

b)

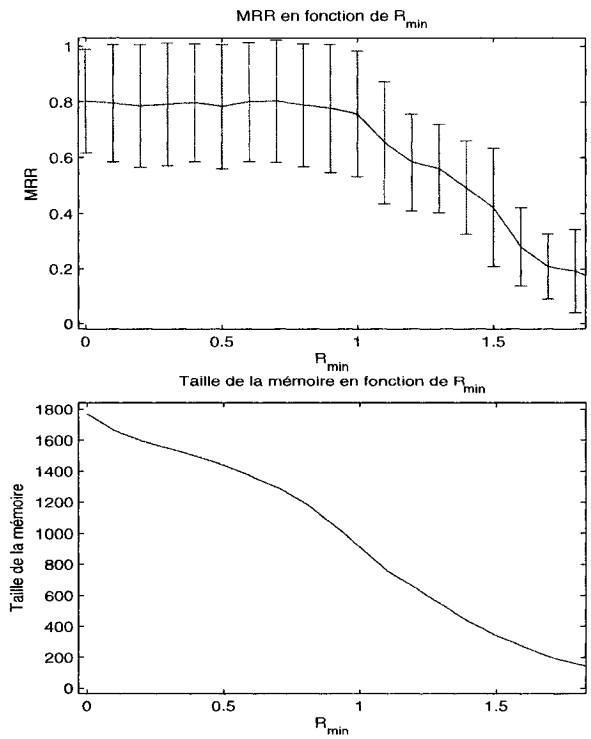

FIG. 6.6 Distance minimale R_{min} entre deux états $\vec{x}_i$ a) MRR en fonction de R_{min}. b) Taille de la mémoire en fonction de R_{min}.

son, ce qui, dans certain cas, diminue l'efficacité du classement. En effet, si la même chanson est détectée plusieurs fois dans la recherche des k plus proches voisins, cela empêchera le système de trouver des thèmes différents pouvant ressembler au thème recherché.

6.4.2 Nombre de plus proches voisins recherchés

On présente, à la figure 6.7, le MRR en fonction du nombre k de plus proches voisins recherchés dans la mémoire.

Plus le nombre k de plus proches voisins recherchés est petit, plus la recherche sera rapide. Pour notre système, on a choisi de rechercher les $k = 3$ plus proches voisins.

a)

b)

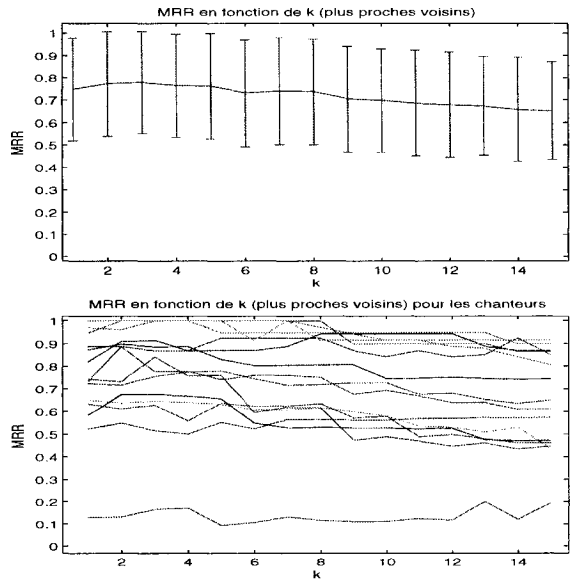

FIG. 6.7 Nombre de plus proches voisins a) MRR en fonction de k pour les 15 chanteurs. b) MRR en fonction de k.

	Apprentissage	Reconnaissance
Sans arbre	37 s	188 s
Arbre métrique	18.9 s	129 s
Arbre hybride avec projections	15,9 s	10,8 s
Arbre hybride sans projections	6,5 s	1,33 s

TAB. 6.2 Tableau du temps d'apprentissage des 36 chansons et du temps de reconnaissance des 130 extraits en fonction du type de recherche effectué. Dans le cas de l'arbre hybride sans projections, on observe une chute du MRR de 20 %.

6.4.3 Arbres de recherche

À l'aide des arbres de recherche, il est possible de diminuer considérablement le temps de recherche des plus proches voisins. On donne au tableau 6.2 les temps de construction et de recherche des différents types d'arbres de recherche utilisés.

Si l'on n'utilise pas d'arbre de recherche, la phase d'apprentissage du système (temps pour écrire les 36 chansons dans la mémoire) prend $37s$[3] et la reconnais-

[3]Avec un Pentium 4 $2,8GHz$.

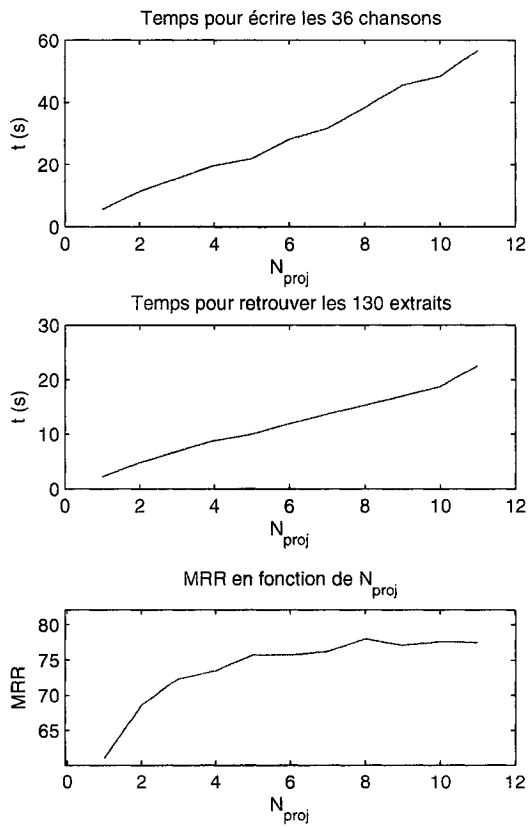

FIG. 6.8 Performances de la recherche (MRR et temps) en fonction du nombre de projections.

sance des 130 extraits prend $188s$. Avec un arbre métrique, on obtient de meilleurs résultats, mais l'accélération n'est pas très grande. En effet, avec un arbre métrique utilisé dans un espace de haute dimension, on se trouve à explorer une grande partie de l'arbre pour s'assurer que les différents nœuds ne contiennent pas de plus proches voisins potentiels. Si l'on utilise un arbre hybride sans projections, on obtient un temps d'apprentissage de $6,5s$ et un temps de reconnaissance de $1,33s$, mais le MRR chute de 20%. La solution idéale est donc d'effectuer une série de projections en basse dimension avec plusieurs arbres hybrides de recherche. On observe alors

qu'à partir de 8 projections (voir figure 6.8), le taux de reconnaissance devient optimal et l'on a tout de même une accélération significative.

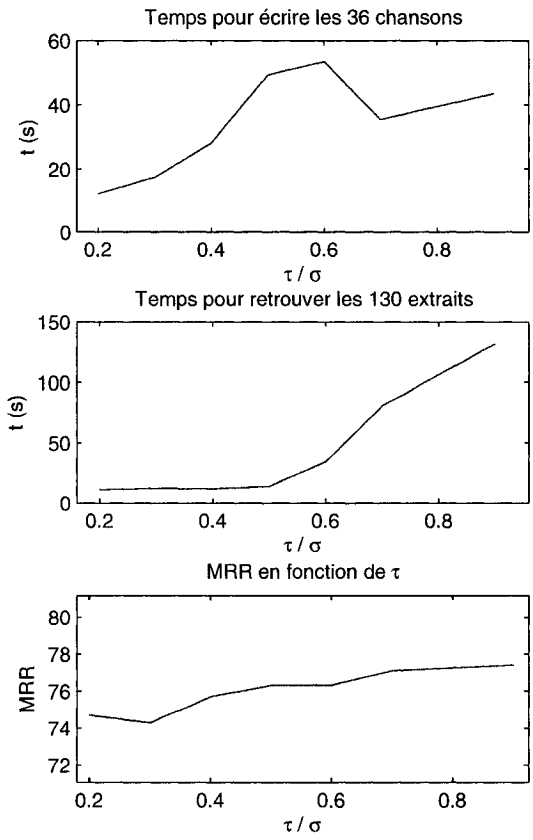

FIG. 6.9 Performances de la recherche (MRR et temps) en fonction de τ.

On a également observé (voir figure 6.9) l'impact de la variation du paramètre τ, l'indice de classement, sur les performances des arbres hybrides. Ici, τ est donné à l'aide de σ, la variance des points dans un espace de dimension N (voir section 5.2.1). Le nombre de projections est fixé à 6. On observe qu'à partir de $\tau = 0,5$, le temps de recherche commence à augmenter de manière significative. Ceci est provoqué par le fait que lorsque τ est grand, un grand nombre de points se retrouvent dans la zone commune aux deux points de pivots (plus de la moitié des points)

	MRR	1^{re} position	n^{bre} d'extraits
Globals	81 %	74 %	130
Chantés	76 %	66 %	44
« ta ta »	82 %	78 %	55
Sifflés	85 %	78 %	31

TAB. 6.3 Tableau du MRR et du pourcentage d'extraits retrouvés en première position pour les différents modes de chant.

et l'on reconstruit alors le nœud comme un nœud métrique. Ceci fait en sorte que l'arbre hybride tend vers un arbre métrique lorsque τ augmente.

La technique de recherche des plus proches voisins retenue utilise un arbre hybride avec $\tau = 0.4$ et on effectue 8 projections vers des espaces de basse dimension (15D).

6.5 MRR pour les différents modes de chant

Un autre test intéressant à effectuer sur le système de reconnaissance de mélodies musicales consiste à comparer les différents modes de chant. Pour ce faire, on a recueilli un ensemble d'extraits mélodiques : chantés librement, en faisant des « ta ta ta » et en sifflant (voir tableau 6.3).

On a observé que la contrainte de faire chanter les participants en faisant des « ta ta ta » améliorait beaucoup l'efficacité de la segmentation des notes. On peut toutefois noter, dans certains cas, que les participants rencontrés avaient plus de difficulté à se rappeler de la mélodie s'ils ne prononçaient pas les paroles. Pour ce qui est des airs sifflés, ils sont généralement faciles à segmenter en notes et ne posent pas trop de problèmes de reconnaissance. Le seul problème étant les chanteurs ayant de la difficulté à siffler de façon continue.

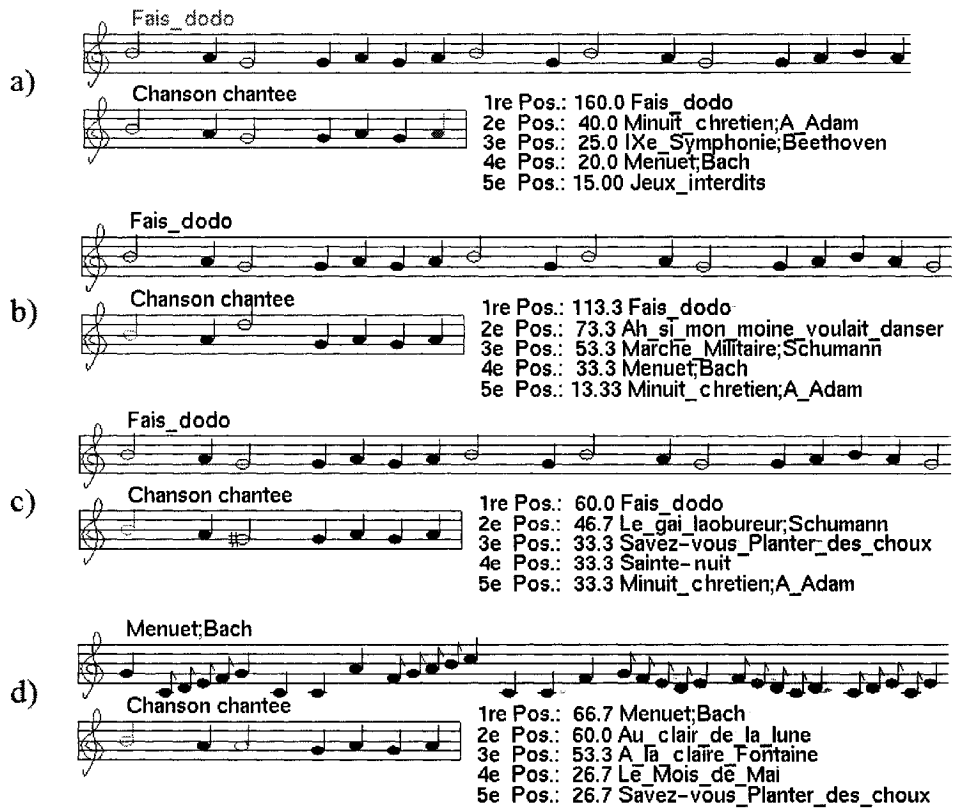

FIG. 6.10 Résultats de recherche pour la chanson Fais dodo. a) Extrait sans erreur. b) 3^e note une quinte trop élevée. c) 3^e note fausse (chantée 1 demi-ton trop aigu). d) 3^e note 2 demi-tons trop aigus.

6.6 Variations mélodiques

Comme nous l'avons vu à la section 3.2, l'emploi de la corrélation harmonique pour définir les corrélations entre les vecteurs d'intervalle mélodique nourrissant le ESN permet de tenir compte de l'harmonicité de la mélodie. On a fait plusieurs tests qui sont présentés à la figure 6.10.

Comme on peut l'observer, des erreurs comme chanter 1 demi-ton à côté ou chanter une variante mélodique en prenant une quinte trop élevée n'empêchent pas de trouver la mélodie désirée. Cependant, si l'on chante 2 demi-tons à coté (ce qui ne

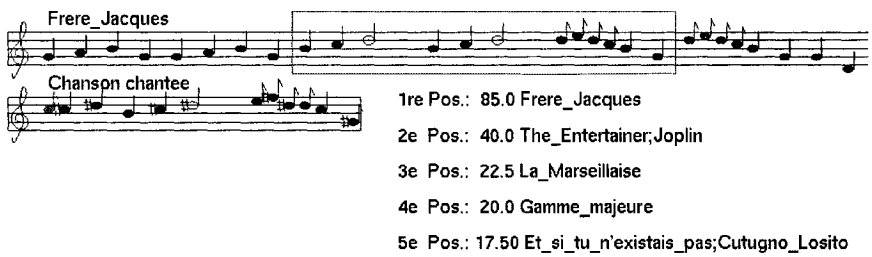

FIG. 6.11 Reconnaissance de la chanson Frère Jacques débutée à la 3^e portée.

devrait pas être une fausse note ou une variante mélodique), l'indice de classement sera moins bon.

Bien que le chant de variantes mélodiques puisse se présenter à l'occasion, les extraits recueillis chez les participants ne présentaient pas de variantes mélodiques. Ceci fait en sorte que considérer des distances linéaires plutôt que tenir compte de la corrélation harmonique entre les intervalles mélodiques donnait de meilleurs résultats. On a obtenu avec les extraits recueillis un MRR global de $0,80$ avec les intervalles mélodiques respectant les corrélations harmoniques, tandis que le MRR est de $0,82$ si l'on ne considère pas la corrélation harmonique (on a alors utilisé une corrélation diminuant de moitié pour chaque distance de un demi-ton). Les erreurs étant plus souvent des fausses notes que des variantes mélodiques, les distances linéaires entre les intervalles mélodiques donnent de meilleurs résultats.

6.7 Début arbitraire

Pour tester le programme, nous avons également tenté de débuter un extrait musical à un instant arbitraire d'une mélodie. À la figure 6.11, on observe que débuter la mélodie Frère Jacques à la 3^e portée n'empêche pas de trouver la mélodie. La faible influence du début de la mélodie est une conséquence de la mémoire évanescente du réseau à écho qui permet d'identifier les schémas musicaux à court terme.

6.8 Conclusion du chapitre

Dans ce chapitre, nous avons vu les performances du programme de reconnaissance de séquences musicales. Cependant, ces performances ne tiennent pas compte de l'interactivité que présente le programme de reconnaissance de séquences. En utilisant le programme, on est plus apte à évaluer la justesse des sons émis puisque l'on visualise directement la séquence des notes interprétées par le programme. Il est de plus possible de réécouter la séquence interprétée et de la modifier. Cela permet, bien entendu, d'améliorer grandement les performances de la recherche.

Pour la segmentation des notes, il est évident que si l'on contraint un chanteur à émettre des syllabes facilement détachables telles que « ta » ou « pa », il est aisé de segmenter les notes. Dans ce cas, on observe nettement une discontinuité dans l'intensité et la fréquence fondamentale du signal. Par contre, lorsque le chanteur chante librement, on n'observe pas toujours des discontinuités facilement identifiables dans l'intensité du signal ou la variation de la fréquence fondamentale. Dans ce cas, il serait judicieux d'observer plus en détail la variation du spectre du son émis. On pourrait alors identifier les différents phonèmes constituant les paroles d'une chanson. Ces dernières pourraient permettre une meilleure segmentation des notes musicales.

Finalement, pour avoir de meilleurs résultats, il faudrait augmenter la collection de chansons pour observer l'impact du nombre de chansons sur les performances du programme de reconnaissance de séquences musicales.

CONCLUSION

Dans cette recherche, nous avions comme objectif la réalisation d'un programme informatique de reconnaissance de séquences musicales fournies par le chant. Pour ce faire, nous avons développé des techniques pour segmenter les notes chantées par l'usager et en extraire la fréquence fondamentale. À l'aide d'un réseau à écho et de la corrélation harmonique, nous avons pu caractériser les séquences mélodiques de la collection de chansons sous la forme d'états internes du réseau à écho. Finalement, l'utilisation d'une mémoire hétéro-associative a permis le stockage et la récupération de titres de chansons associés aux états du réseau à écho.

L'observation des résultats a permis d'optimiser les paramètres de segmentation, les paramètres du réseau à écho et ceux de la mémoire hétéro-associative pour avoir un meilleur classement des extraits recueillis chez les chanteurs. Un problème demeure, c'est celui de la segmentation des notes. En effet, la plupart des extraits n'ayant pas été trouvés en première position ont été mal segmentés. Les notes interprétées n'étaient pas les bonnes. Une solution à ce problème est de forcer les gens à chanter avec des syllabes facilement segmentables (« ta », « da », « pa », etc.). Cependant, il est préférable de laisser le chanteur libre de chanter une chanson telle qu'il se la rappelle. On peut alors penser utiliser la prédiction linéaire (LPC) pour aider à la segmentation comme on l'utilise en reconnaissance de la parole (Rabiner et Juang, 1993). Cette alternative a été essayée, mais elle ne semblait pas donner de résultats encourageants.

Dans notre travail, nous avons présenté comment il était possible de représenter les intervalles mélodiques sous une forme qui respecte la distance relative entre les spectres des sons composant les intervalles à l'aide de la corrélation harmonique. Ceci a donné les vecteurs d'intervalle mélodique VIM qui étaient présentés à l'entrée

du réseau à écho. Cependant, bien que l'utilisation de la corrélation harmonique permettait de tenir compte de variations harmoniques, les résultats obtenus étaient meilleurs lorsque l'on utilisait une simple corrélation diminuant de moitié pour chaque distance de un demi-ton.

Dans notre méthode, l'enregistrement des séquences musicales sous forme d'états du réseau à écho simplifie l'apprentissage. Il n'est pas nécessaire de faire l'apprentissage de neurones de sortie. De plus, il n'est pas nécessaire non plus d'extraire les différents thèmes des chansons de la collection. Par exemple, le projet MUSART mentionne avoir utilisé une banque de 2 844 thèmes extraits de 258 chansons des Beatles. Cela donne environ 11 thèmes par chanson. Dans notre cas, nous avons utilisé une collection de 36 chansons à laquelle nous avons associé 1 200 états internes du réseau à écho. Cela donne une moyenne de 33 états par chanson. Il n'est donc pas nécessaire d'identifier les thèmes puisqu'ils sont inscrits implicitement dans la mémoire du réseau à écho. De plus, les thèmes semblables occupent le même espace mémoire dans la mémoire associative, ce qui économise de l'espace.

Pour améliorer cette recherche, il faudrait penser augmenter le nombre de titres de la collection de chansons et le nombre de participants. Ceci permettrait d'améliorer la fiabilité des résultats. On pourrait alors voir si notre système utilisant un réseau à écho serait affecté de la même manière qu'un système utilisant les modèles de Markov ou l'alignement de contours. Comme l'ont démontré Dannenberg et al., les performances d'un système de reconnaissance de séquences musicales (Dannenberg et al., 2003) tendent à diminuer dramatiquement lorsque la collection de chansons devient importante. Il faudrait comparer plus rigoureusement notre méthode avec les autres méthodes existantes en utilisant une banque de données unique. En effet, il s'est avéré difficile de comparer les résultats, car ils varient grandement en fonction de plusieurs facteurs (chanteurs, taille de la banque de chansons, types de chansons, contraintes des chanteurs, etc.). On peut trouver par exemple dans les différentes

études des MRR allant de 13,4 % à 90 %.

Un autre point sur lequel il serait bon de porter attention, c'est l'inégalité des chances de récupération des différentes mélodies. En effet, certaines mélodies ont un contenu plus varié et contiennent un plus grand nombre de thèmes. Elles ressortent plus souvent lors de la recherche puisqu'il y a plus de chance que leurs thèmes ressemblent aux thèmes chantés. De plus, certaines chansons étant plus populaires, il serait peut-être opportun de donner plus de poids à ces dernières ou encore de penser séparer la banque de données en différents styles musicaux.

On peut également penser effectuer davantage de tests sur la corrélation harmonique. Par exemple, on pourrait effectuer une série de tests systématiques et rigoureux en transposant des sections d'extraits de r demi-tons et observer l'impact de la transposition sur les performances de récupération de la mélodie.

Pour la représentation des intervalles mélodiques sous forme de VIM, il serait possible d'explorer d'autres applications. On pourrait par exemple imaginer mesurer la ressemble intrinsèque entre diverses chansons de la banque de données. On pourrait également imaginer réaliser un système permettant la composition automatique de variantes mélodiques en choisissant des intervalles mélodiques probables dans une séquence musicale.

Finalement, on pourrait penser à une autre approche pour la reconnaissance de séquences musicales. Celle-ci consisterait à observer la fréquence fondamentale en temps continu. On pourrait alors utiliser le réseau à écho en mode continu (Jaeger, 2001) plutôt qu'en mode discret. Dans ce cas, on fournirait en continu les intervalles mélodiques observés en se basant sur le tempo détecté. Ceci pourrait éviter les problèmes de segmentation qu'on a rencontrés.

RÉFÉRENCES

ACHLIOPTAS, D. (2001). "Database-friendly random projections", 20th Annual Symposium on Principles of Database Systems, Santa Barbara, CA, p. 274-281.

ADAMS, N. H. (2004). "Time Series Representations for Music Information Retrieval", Rapport technique du laboratoire CSPL n^o TR 349, Université du Michigan, citeseer.ist.psu.edu/adams04time.html.

BISHOP, C. M. (1995). "Neural Networks for Pattern Recognition", Oxford University Press, New York, 482 p.

COVER, T. M. (1965). "Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition", IEEE Transactions on Electronic Computers, volume EC-14, p. 326-334.

DANNENBERG, R., W. BIRMINGHAM et G. TZANETAKIS. (2003). "The MUSART Testbed for Query-By-Humming Evaluation", Proceedings of ISMIR 2003, Baltimore.

HAYKIN, S. (1999). "Neural Networks : A Comprehensive Foundation", Second Edition, Prentice Hall, New Jersey, 842 p.

JAEGER, H. (2001). "The echo state approach to analysing and training recurrent neural networks", Rapport technique GMD n^o 148, Franhofer Institute for Autonomous Intelligent Systems.

LARTILLOT, O. (2003). " Discovering Musical Patterns through Perceptive Heuristics", Proceedings of ISMIR 2003, Baltimore, ismir2003.ismir.net/papers/Lartillot.PDF.

LIU, T., A. W. MOORE, A. G. GRAY et K. YANG (2004). "An Investigation of Practical Approximate Nearest Neighbor Algorithms", Proceedings of NIPS 2004, Vancouver.

MEEK, C. et W. BIRMINGHAM (2002). "Johnny Can't Sing : A Comprehensive Error Model for Sung Music Queries", Proceedings of ISMIR 2002, p. 124-132.

PACHET, F. et J.-P. BRIOT (2004). *Informatique musicale*, Hermes-Science, Paris, 441 p.

PARDO, B., J. SHIFRIN et C. MEEK (2002). "HMM-based musical query retrieval", JCDL 02 : Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries, ACM Press, New York, p. 295-300.

RABINER, L. et B.-H. JUANG (1993). "Fundamentals of Speech Recognition", Prentice Hall, New Jersey, 507 p.

Théorie de la musique, neuvième édition (1982). École de musique Vincent-d'Indy, 170 p.

ANNEXE I

COLLECTION DE CHANSONS

1. Fais dodo
2. Jingle Bells
3. Au clair de la lune
4. IX^{e} Symphonie de Beethoven
5. Mary Had a Little Lamb
6. Sonate 331 de Mozart
7. Le mois de mai
8. J'ai du bon tabac
9. Frère Jacques
10. À la claire fontaine
11. Il était un petit navire
12. V'là l'bon vent
13. Ah si mon moine voulait danser
14. Le Roi Dagobert
15. Savez-vous planter des choux ?
16. When The Saints Go Marching In
17. Isabeau
18. Menuet de Bach
19. Mon beau sapin
20. Jeux interdits

21. Sainte nuit
22. Le Parrain
23. Mickael chante avec moi
24. C'est la belle Françoise
25. La Marseillaise
26. Marche militaire de Schumann
27. Le gai laboureur de Schumann
28. Minuit chrétien de A. Adam
29. Happy Birthday to You
30. Pour Élise de Beethoven
31. Old 100th de Psalter
32. Noble époux de Marie de Dufour
33. Menuet de Beethoven
34. The Entertainer de Joplin
35. Et si tu n'existais pas de Cutugno et Losito
36. Gamme majeure